

Знакомство с Unity

Занятие 1

Оглавление

ВВЕДЕНИЕ	3
УСТАНОВКА UNITY HUB и UNITY EDITOR.....	4
СОЗДАНИЕ ПРОЕКТА.....	8
ГОРЯЧИЕ КЛАВИШИ.....	11
ЗНАКОМСТВО С UNITY EDITOR	12
НАВИГАЦИЯ ПО СЦЕНЕ.....	19
ИГРОВЫЕ ОБЪЕКТЫ.....	20
КОМПОНЕНТЫ	21
ПРЕФАБЫ.....	23
СБОРКА ПРОЕКТА	24
ПРИЛОЖЕНИЕ 1. ТУТОРИАЛЫ К 2D PLATFORMER	25
ПРИЛОЖЕНИЕ 2: ДОБАВЛЕНИЕ ASSETS И PACKAGES	32
ДОМАШНЕЕ ЗАДАНИЕ	35
ИНТЕРЕСНЫЕ ПРОЕКТЫ	36
СПИСОК ЛИТЕРАТУРЫ	37
УЧЕБНАЯ КАРТА ДИСЦИПЛИНЫ.....	38

ВВЕДЕНИЕ

Рекомендуем сразу перейти на [сайт Unity](#) и зарегистрироваться на нем, создав Unity ID (кнопка в правом верхнем углу сайта). Без наличия Unity ID пользование экосистемой Unity не представляется возможным.

Общий список полезных платформ Unity:

- [Unity Learn](#): платформа с обучающими туториалами и различными уроками.
- [Unity Learners Group](#): доска объявлений для всех, кто учит Unity.
- [Unity Microgames](#): пошаговые инструкции для новых пользователей. Обучение работе в редакторе и интересные туториалы.
- [Learn tab in Unity Hub](#): доступ к различным обучающим ресурсам.
- [Unity Documentation](#): мануал.
- [Unity Answers](#) или [Unity Forums](#): форум для того, чтобы задать вопрос, или найти ответы.
- [Unity YouTube Channel](#).
- [Unity Blog](#): новости и т.п.
- [Asset Store help](#).

Официальная документация по Unity 2020.3:

- [На английском](#)
- [На русском \(2020.2\)](#)

Список полезных книг по Unity, а также неплохие проекты с инструкциями можно найти в конце данного файла.

[В ОГЛАВЛЕНИЕ](#)

УСТАНОВКА UNITY HUB и UNITY EDITOR

Начать нужно прежде всего с установки Unity Hub и Unity Editor.

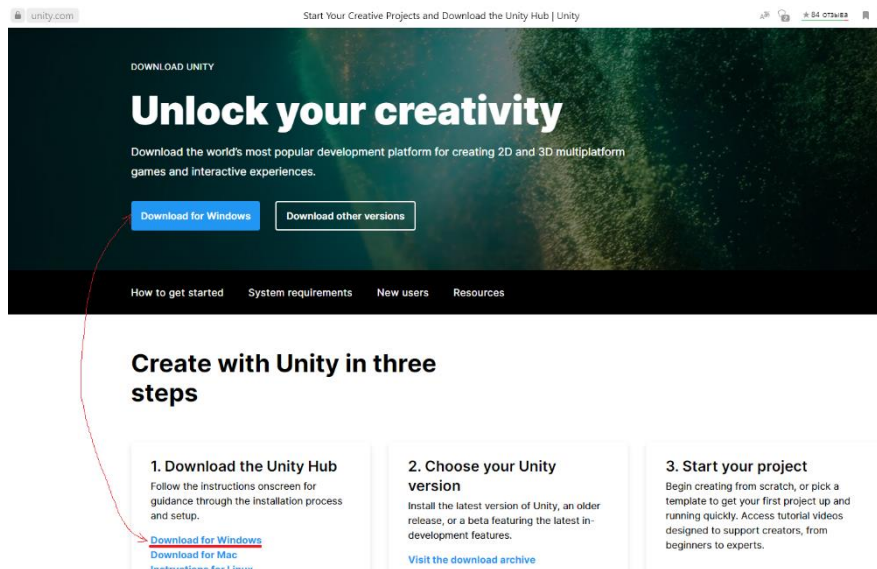
Пояснение к этой процедуре есть на платформе Unity Learn как [отдельный курс обучения](#). Рекомендуем начать с него.

Выбираем справа вверху версию 2020.3 и следуем инструкциям.

Шаг 1. Нажимаем на кнопку **Mark step as completed**.

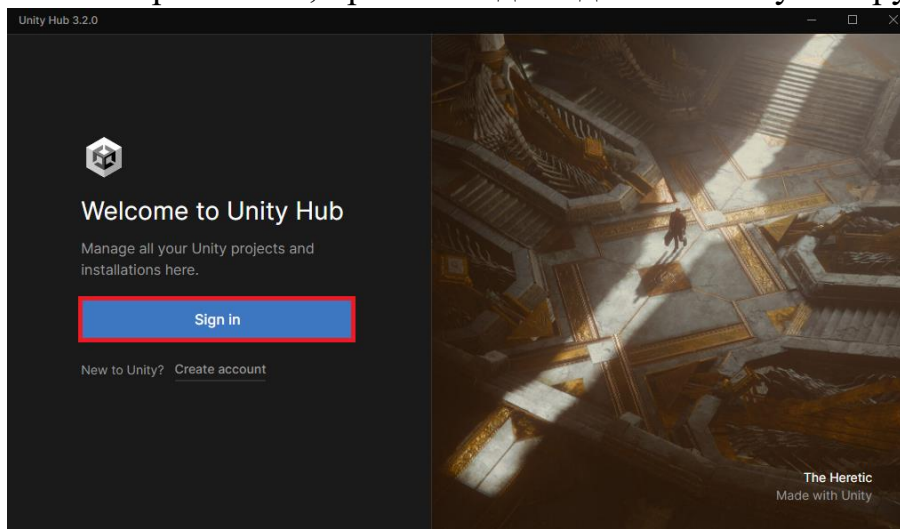
Шаг 2. Знакомимся с [системными требованиями](#).

Шаг 3. Переходим на [страницу скачивания](#). Нажимаем [Download for Windows](#) (вверху или внизу).

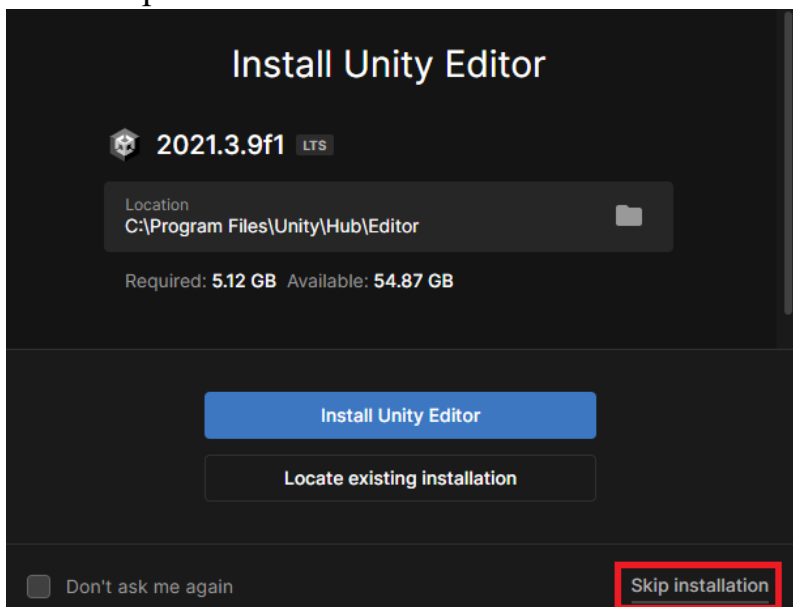


Шаг 4. Устанавливаем скачанный файл **UnityHubSetup.exe**. По завершении установки можно оставить галочку на пункте **Запустить Unity Hub**.

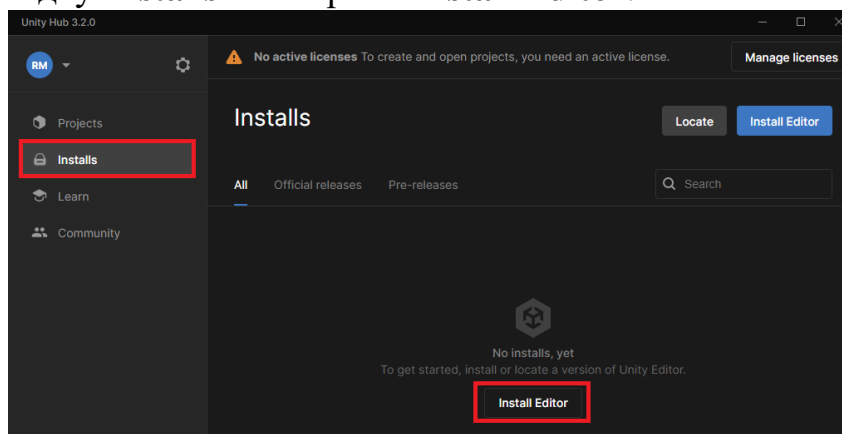
При открытии Unity Hub система попросит вас залогиниться или создать аккаунт. Выбираем **Sign in**. Должен произойти автоматический вход в ваш ранее созданный аккаунт. Если этого не произошло, просто введите данные аккаунта вручную.



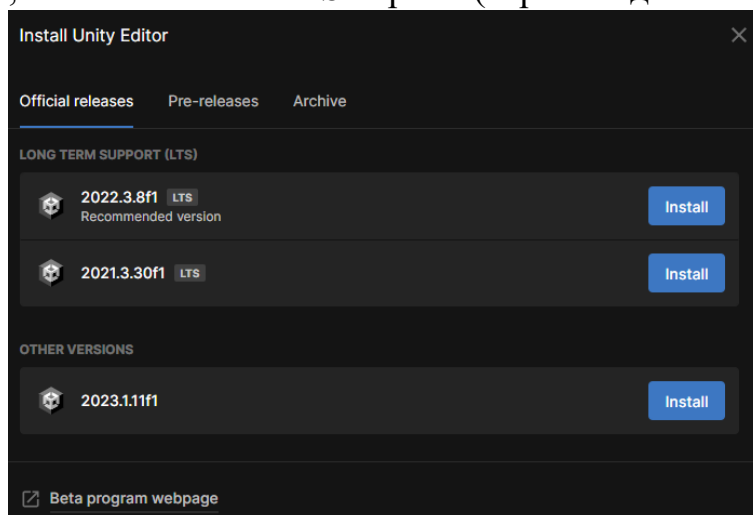
При первом запуске Unity Hub будет навязывать вам конкретную версию Unity Editor. Выберите пункты **Skip installation** и **Skip license activation**, чтобы отказаться от установки предлагаемой версии.



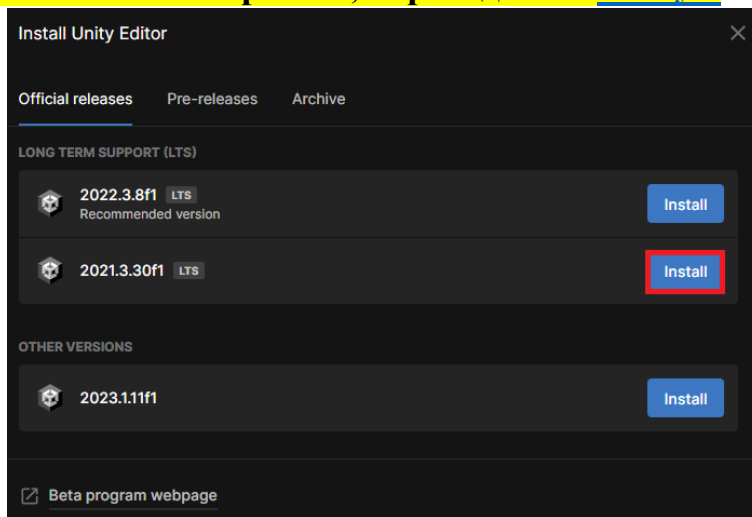
Переходим на вкладку **Installs**. Выбираем **Install Editor**.



Для работы над вторым заданием (2D Arcanoid) рекомендуем использовать версию **2020.3.16f1**. Однако, ее нет в списке LTS-версий (версии с длительной поддержкой).

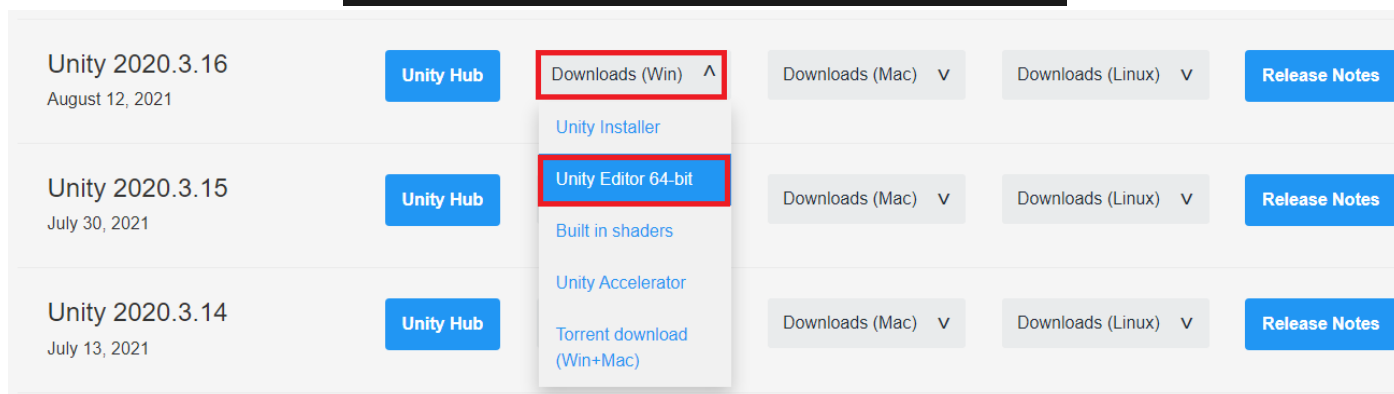
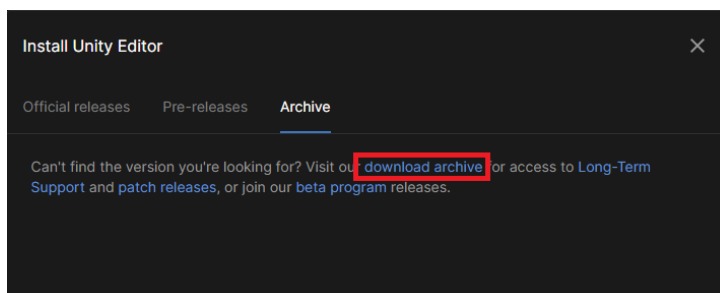


В открывшемся окне из LTS-версий наиболее близкая к нашей – версия **2021.3.30f1** (последний релиз версий 2021-го года). Можно установить более новую версию (возможны незначительные отличия в интерфейсе!). Если вы выбрали новую версию, на этом **работа по установке завершена, переходите к Шагу 5.**



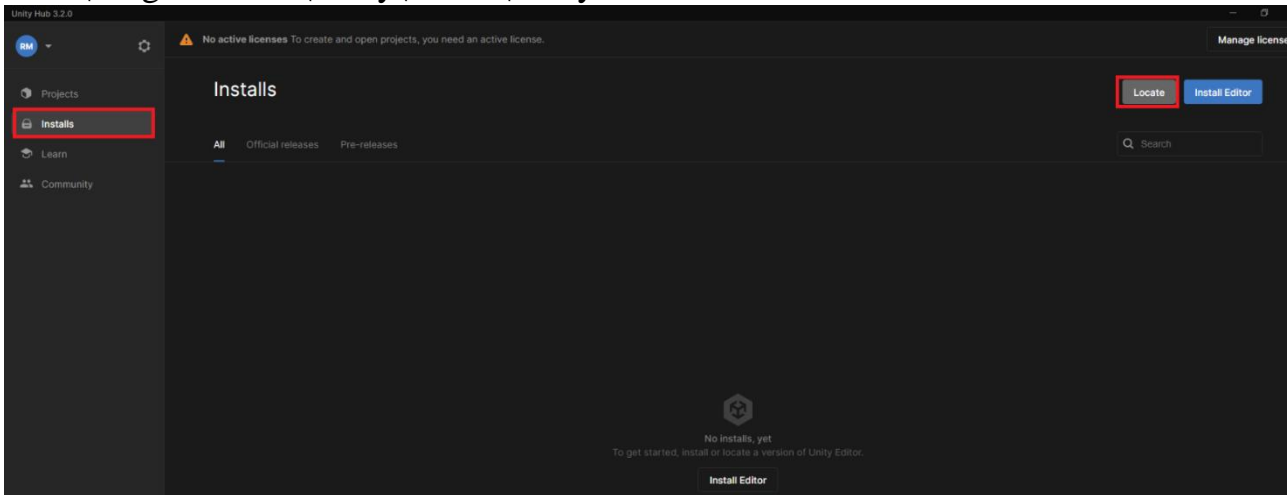
Замечание. При установке версии **2021.3.30f1** Unity спросит вас, какие модули вы хотите установить в дополнение к редактору. Необязательно отмечать всё и сразу, но желательно на всякий случай охватить как можно больше платформ.

Если вы хотите установить непосредственно версию **2020.3.16f1** (будет ограничен некоторый функционал), то надо щелкнуть по вкладке **Archive**, а затем по надписи **download archive**. В открывшемся окне выбрать **Unity 2020.x**, затем найти искомую версию **2020.3.16f1**, выбрать **Downloads (Win)** → **Unity Editor 64 bit**. Файл объемный – весит около 2.4 Гб. Установите его на свой компьютер.

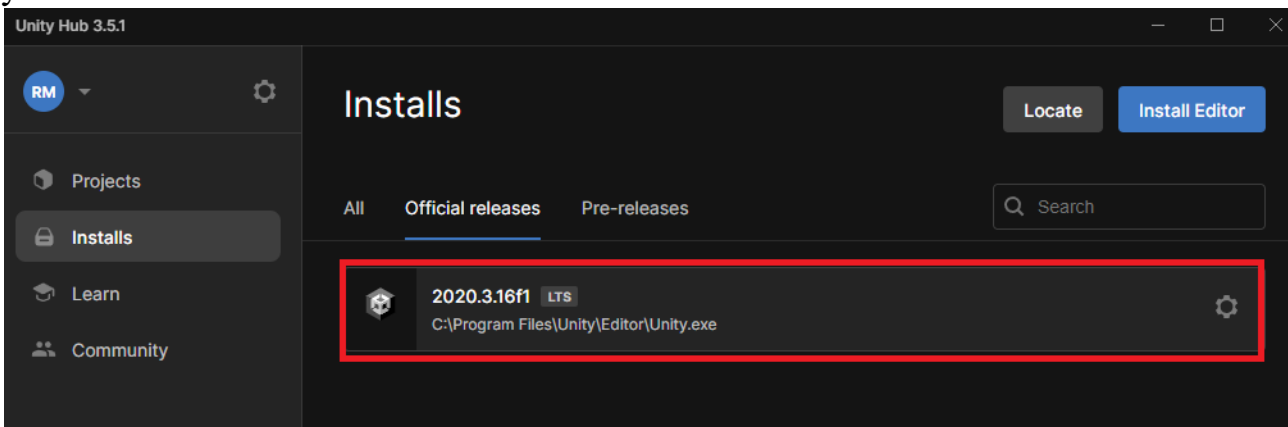


После установки Editor`а, откройте Unity Hub. На вкладке **Installs** скорее всего так и останется пусто. Надо помочь Unity Hub распознать установленную версию Editor`а. Для этого выбираем в правом верхнем углу кнопку **Locate**, чтобы вручную указать Unity Hub на нашу версию редактора.

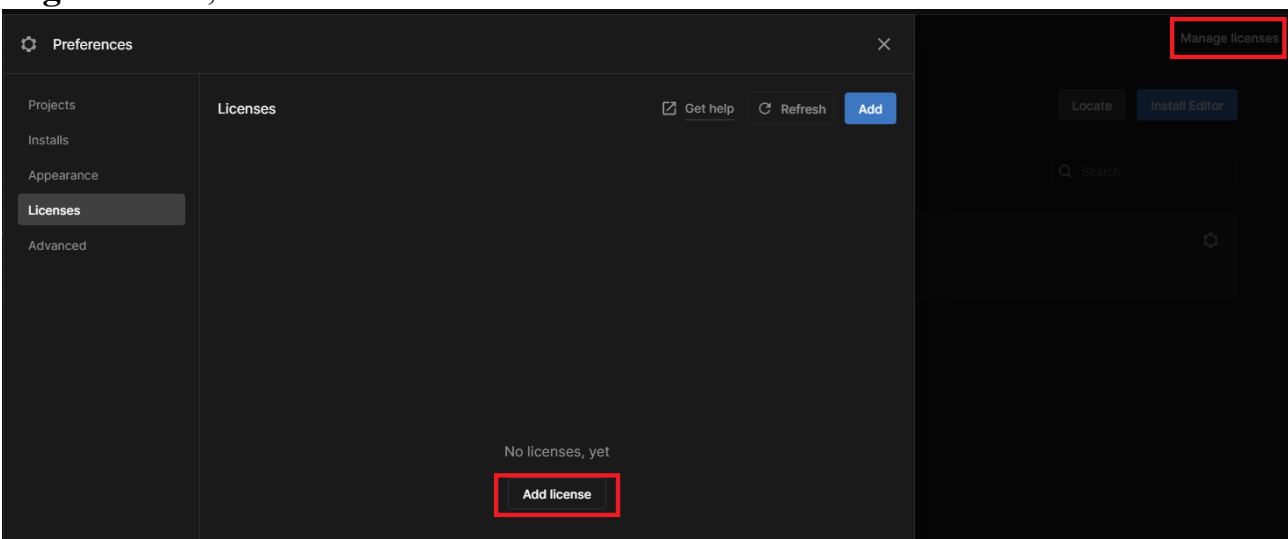
Стандартный путь к exe`шнику редактора:
C:\Program Files\Unity\Editor\Unity.exe



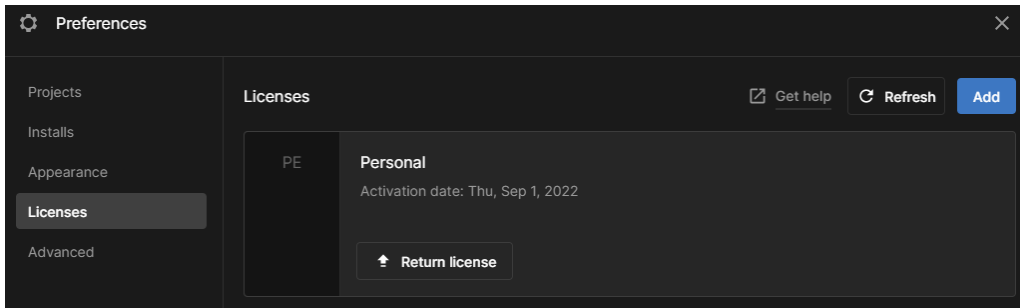
Результат:



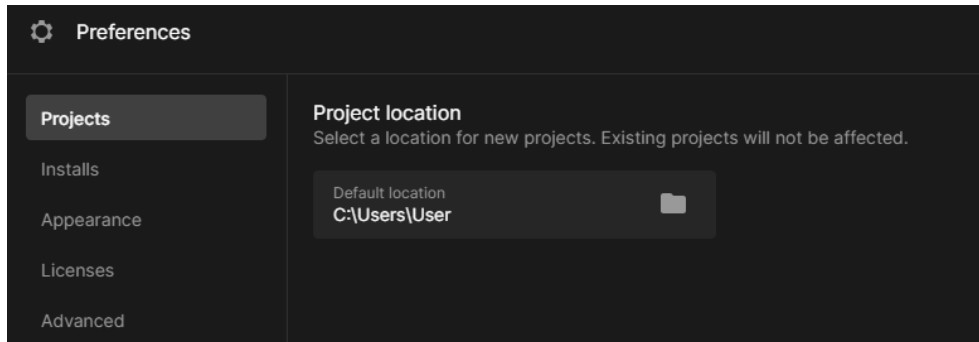
Шаг 5. В верхней части экрана по-прежнему висит назойливое упоминание про то, что на данный момент у нас нет активной лицензии. Пришло время выбрать ее. Нажимаем **Manage licenses**, а затем **Add license**.



Среди лицензий надо найти **Personal**. Активируется персональная лицензия для некоммерческих целей.



На вкладке **Projects** можно регулировать расположение новых проектов, создаваемых в редакторе.



Установка Unity Hub и Unity Editor завершена! Можете «прокликать» 5-минутный курс как заверченный.

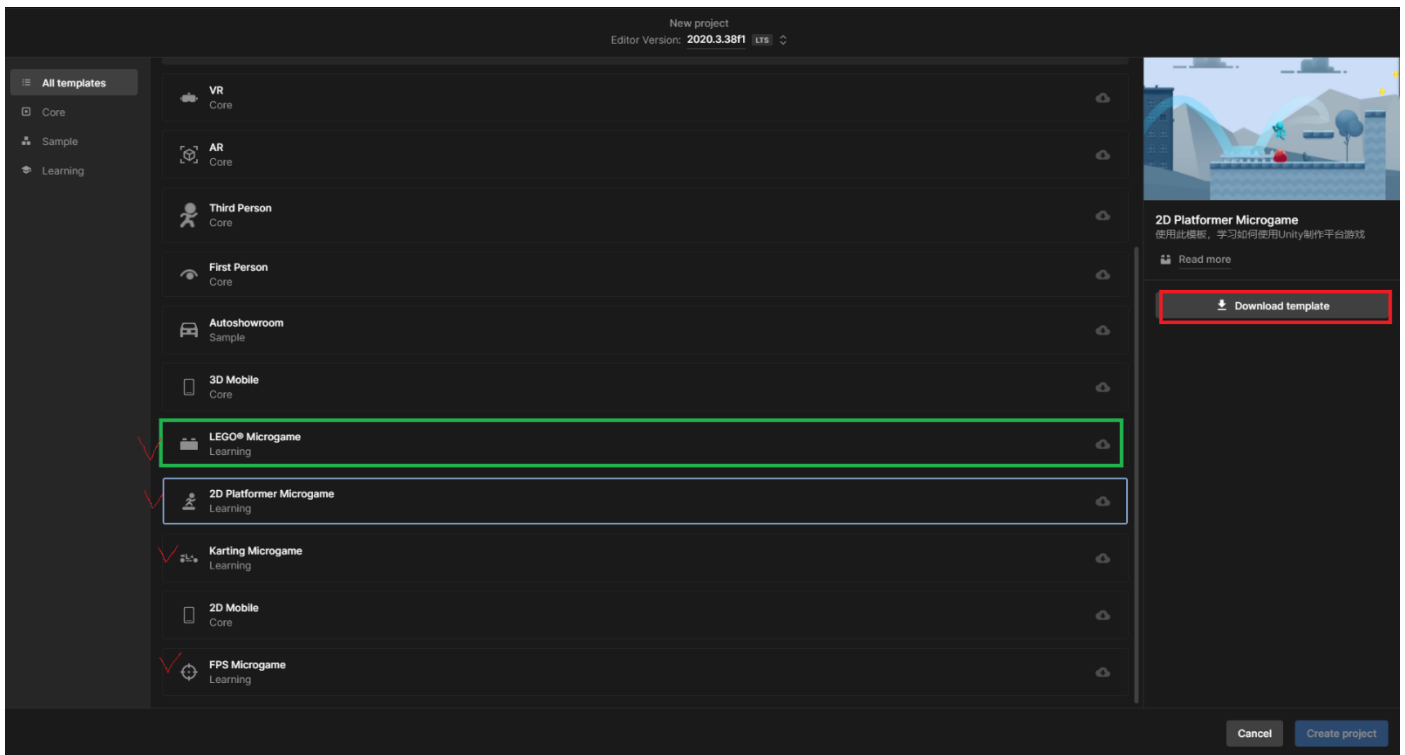
[В ОГЛАВЛЕНИЕ](#)

СОЗДАНИЕ ПРОЕКТА

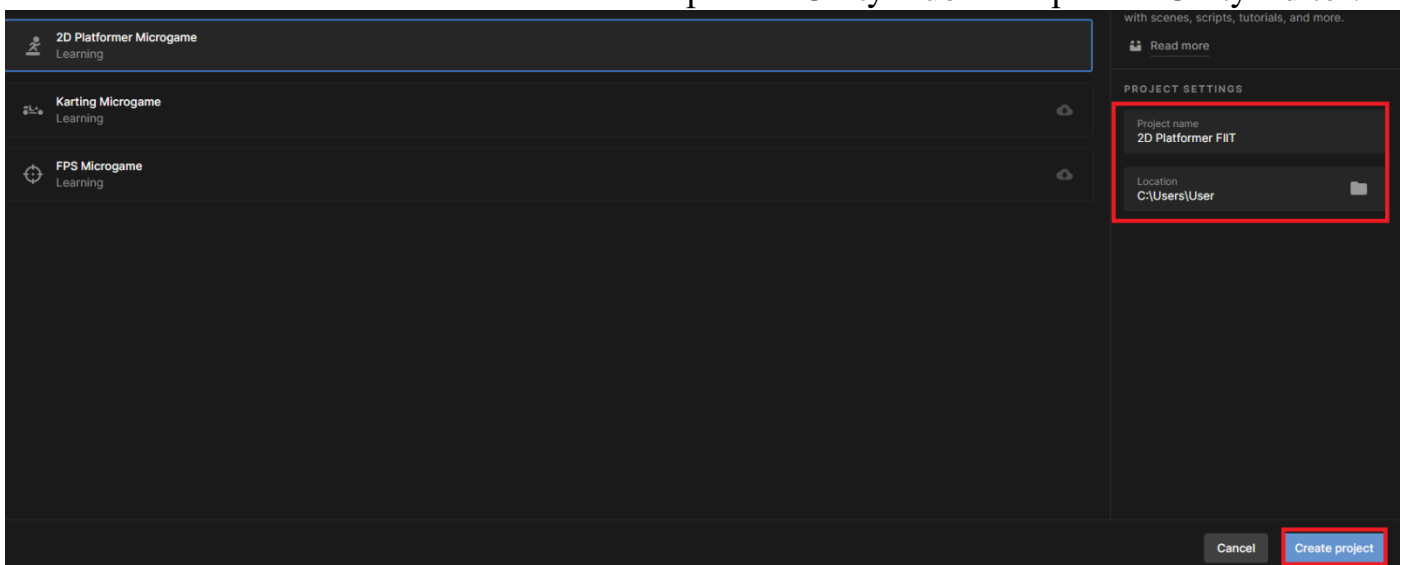
Создание проекта занимает всего несколько минут. Разработчики курсов на Unity Learn предлагают в начале создать шаблонный проект с микроигрой и пройти небольшие туториалы.

Микроигра – это готовая игра со встроенными обучающими программами, которые вы можете настраивать и в которые можно играть. Попутно вы узнаете об интерфейсе редактора Unity, о том, как работать с объектами и как изменить режим сцены.

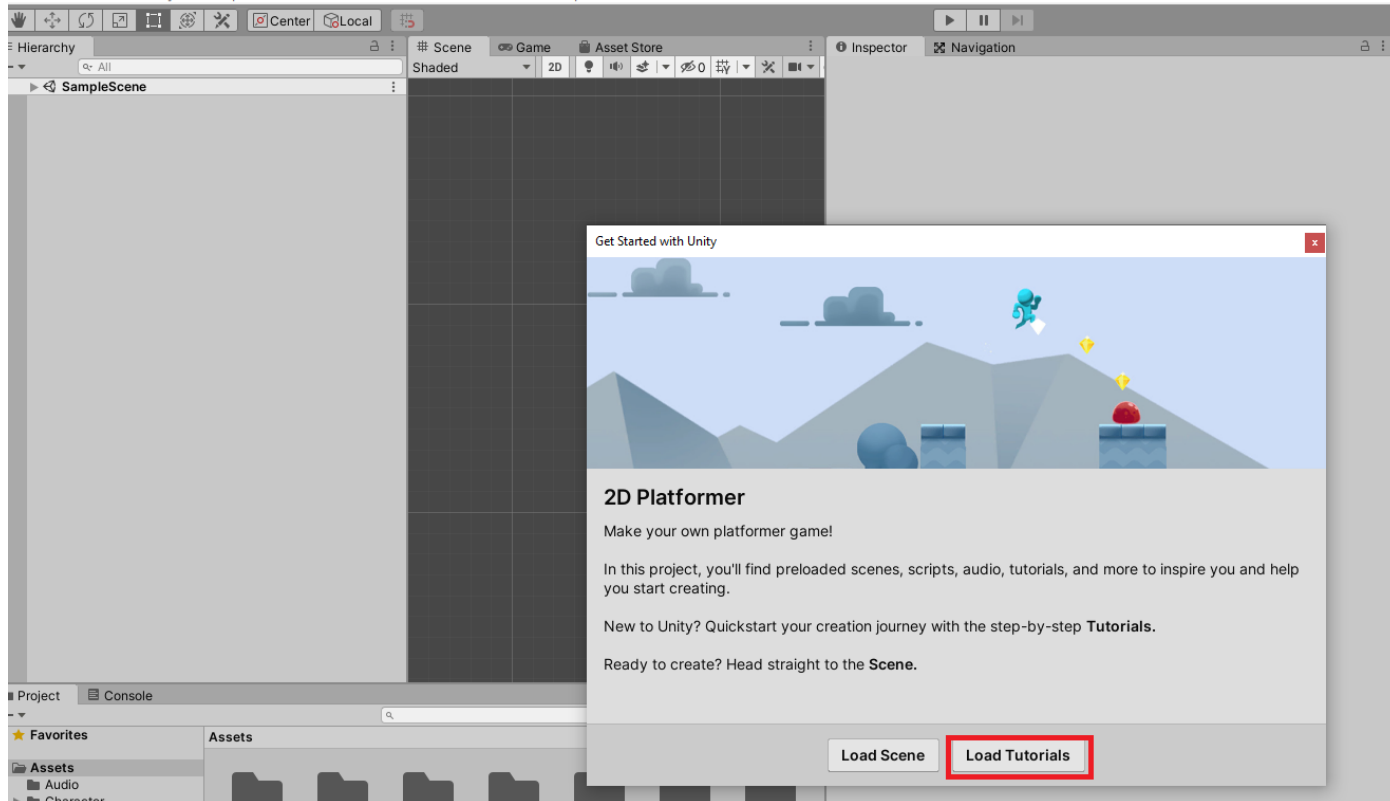
1. Открываем Unity Hub
2. Выбираем вкладку **Projects** слева.
3. В правом верхнем углу нажимаем **New Project**.
4. Откроется большой список (если вы установили много модулей) шаблонов. Для создания обычных 2D или 3D проектов можно просто выбрать один из двух первых пунктов. В данном списке есть четыре базовые микроигры. Выберем **2D Platformer** и нажмем справа на кнопку **Download template**. Необязательно выбирать именно 2D Platformer, **Lego Microgame** тоже весьма полезна и поможет узнать кое-что новое.



5. После скачивания шаблона станет активна кнопка **Create project** в правом нижнем углу. Обратите внимание на расположение вашего проекта и его название! После нажатия кнопки закроется Unity Hub и откроется Unity Editor.



6. В окне Unity Editor Вам будет предложено сразу начать работать на сцене или попробовать tutoriales («Load Tutoriales»). Работа с tutoriales к игре 2D Platformer дана в [Приложении 1](#).



[В ОГЛАВЛЕНИЕ](#)

ГОРЯЧИЕ КЛАВИШИ

Полный список горячих клавиш можно найти [по ссылке](#), здесь перечислены лишь наиболее популярные (для ОС Windows):

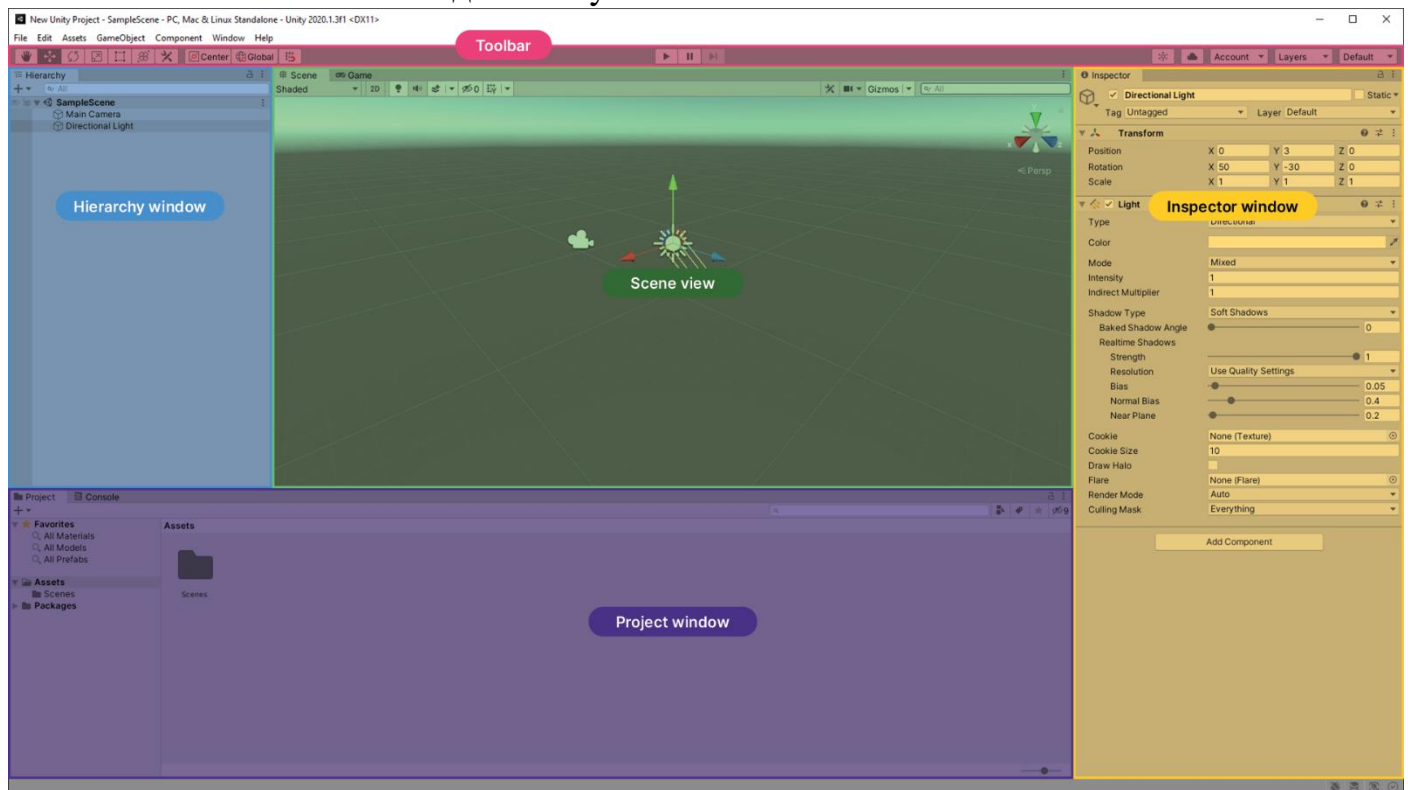
- F – Следует нажать для того, чтобы найти и сфокусироваться на определенном объекте в окне сцены, для этого объект следует предварительно выделить или в окне иерархии, или в окне сцены. Аналогом может служить двойное нажатие ЛКМ.
- Ctrl+D – Одна из наиболее популярных комбинаций клавиш, используется для создания копии объекта или ассета, который в данный момент выделен. Комбинация работает в окне иерархии и окне проекта.
- Q – Pan (перемещение камеры сцены). Аналог – нажатие на колесико мыши.
- W – Move (перемещение объекта). Этот инструмент обычно выбран по умолчанию при выборе объекта.
- E – Rotate (вращение объекта).
- R – Scale (масштабирование).
- Ctrl+ЛКМ – Snap (перемещение и поворот объекта с определённым шагом, который задан в пользовательских настройках).
- Также работают стандартные сочетания клавиш для отмены, копирования, вставки и поиска.
- В меню GameObject на верхней панели редактора находится несколько удобных функций, которые также имеют горячие клавиши, но так как функции специфические и сочетание клавиш трудные для запоминания, проще использовать меню:
 - Set as first sibling – делает выбранный объект первым в иерархии внутри родительского объекта;
 - Set as last sibling – делает выбранный объект последним в иерархии внутри родительского объекта;
 - Move to view – переместить камеру редактора к выбранному объекту;
 - Align with view – переместить выбранный объект в позицию камеры редактора;
 - Align view to selected – переместить камеру в позицию выбранного объекта;
 - Toggle active state – включение/выключение объекта (проще использовать checkbox в окне инспектора).

[В ОГЛАВЛЕНИЕ](#)

ЗНАКОМСТВО С UNITY EDITOR

Можно создать пустой 2D проект для тренировки.

Ознакомимся с внешним видом Unity Editor



Рабочая область в Unity делится на 5 частей: верхний **Toolbar** (тулбар с инструментами), окно **Hierarchy** (иерархия), **Scene view** (сцена), окно **Inspector** (инспектор) и **Project window** (окно проекта).

Scene – окно сцены, в котором непосредственно осуществляется сборка игровых уровней.

Сцену можно представить как один уровень в игре (за редким исключением), поэтому разработанная игра в итоге состоит из множества игровых сцен. Панель управления данного окна позволяет выбирать различные настройки просмотра сцены, влияющие на вид сцены в процессе разработки и не влияющие на создаваемую игру.

Панель управления окна сцены.

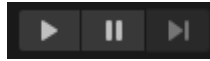


- Первый выпадающий список выбирает, какой режим отрисовки будет использоваться для отображения сцены.
- Переключение между 2D и 3D-режимами.
- Включение освещения и звуков на сцене.
- Второй выпадающий список включение и выключение эффектов: Skybox (текстура, визуализируемая на заднем фоне), Fog (туман), Flares (блики от источников света).
- Включение/выключение сетки.
- Настройки камеры сцены.

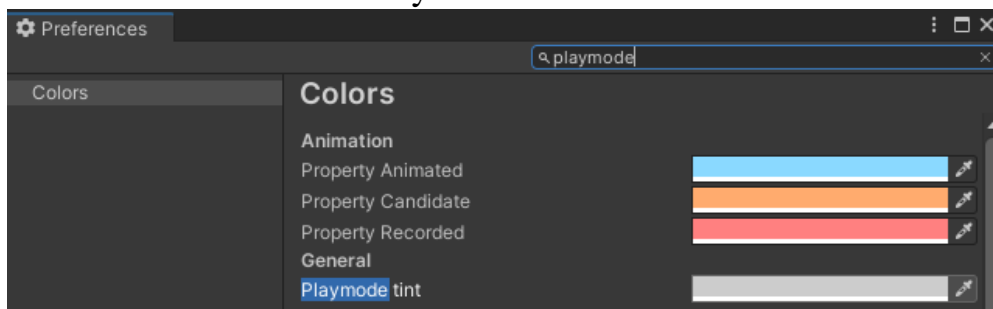
- Меню Gizmos – это графические иконки, которые помогают визуализировать и идентифицировать объекты на сцене (имеются встроенные иконки, но также можно создавать новые для своих объектов).
- Последнее – это поле поиска объектов на сцене.

По центру кроме **Scene view** есть еще **Game view**, где вы будете наблюдать за игрой при ее запуске и тестировании.

Game – окно игры, в котором отрисовывается изображение из камеры (камер) со сцены. Для управления игровым режимом используются кнопки (Запуск, Пауза, Следующий кадр) на панели по центру редактора Unity.



После запуска мы можем изменять положения объектов, параметры компонентов и т.д., все эти значения не сохранятся, когда мы остановим игру, кроме изменений, связанных с ассетами. Чтобы лучше можно было различать в каком сейчас состоянии игра (запущенном или нет), можно настроить изменение цвета интерфейса редактора в меню: Edit – Preferences...Colors – Playmode tint.

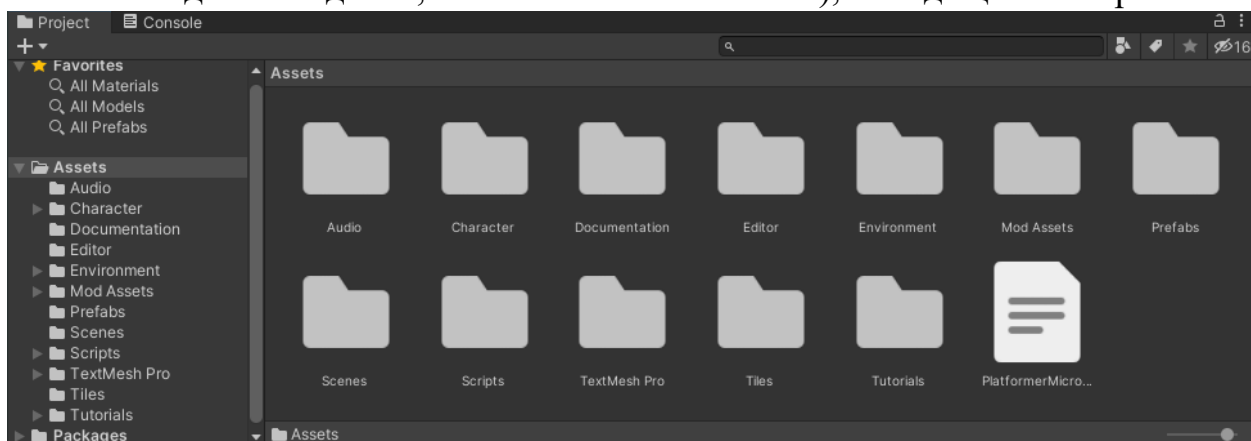


Панель управления игрового окна.



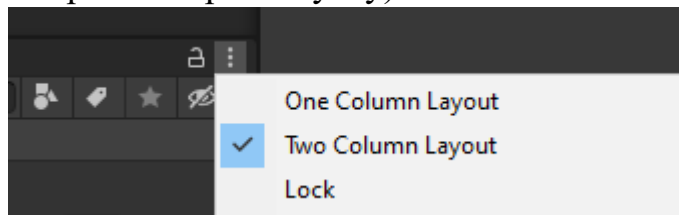
Первый выпадающий список позволяет отображать изображения с различных камер в игре, на различные дисплеи (возможность поддерживается Windows, Mac OS X, Linux). Во втором выпадающем списке устанавливается соотношение сторон или разрешение экрана, чтобы посмотреть, как игра будет отображаться на разных дисплеях. Переключатели: Maximize on Play (игровое окно будет растягиваться на 100% после запуска игры в редакторе), Mute audio (выключение звуков), Stats (окно статистики визуализации, полезно для мониторинга производительности игры), Gizmos (отображение объектов Gizmos в окне игры).

Project window – окно проекта, где отображаются все ассеты (файлы, которые физически находятся на диске, их можно найти в папке), находящиеся в проекте.



Project window работает как файловый менеджер, организованный в папки. Вы можете перетаскивать ассеты прямо из **Project window** на **Scene view**.

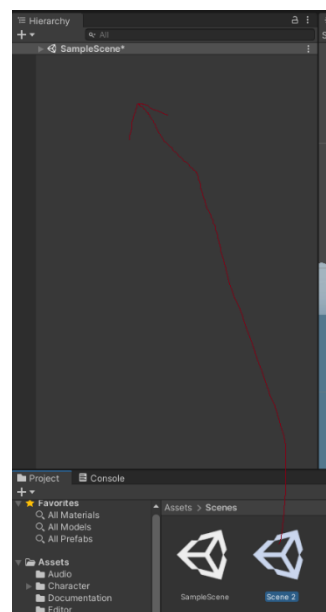
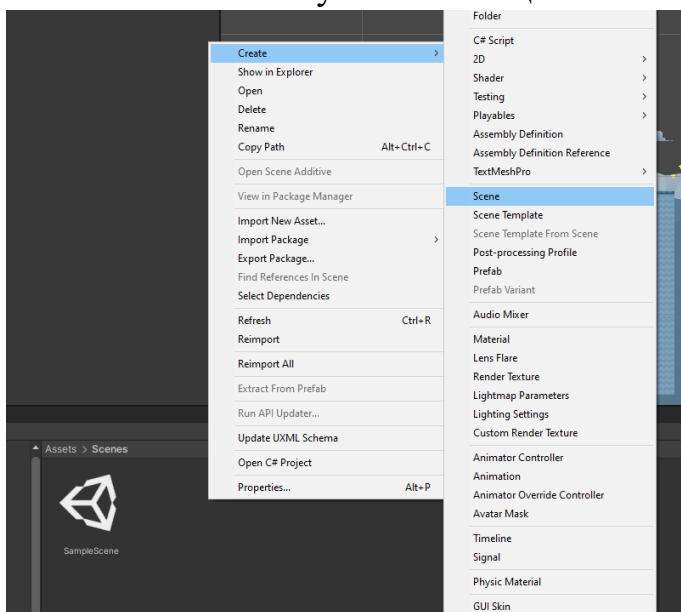
Слева отображается структура проекта в виде иерархического списка, справа отображается содержимое (также можно оставить лишь правую часть, выбрав One Column Layout в меню в правом верхнем углу).



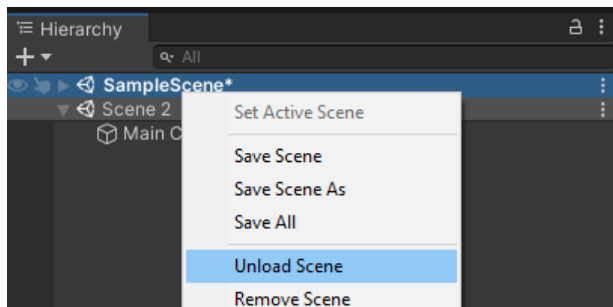
В разделе Favorites можно хранить результаты поиска ассетов в проекте. Окно проекта предоставляет мощные возможности поиска ассетов по имени, типу, метке. Для создания новых ассетов используется кнопка + в левом верхнем углу окна или кнопка Create в контекстном меню. Кроме создания ассетов, в меню можно: показать проект в проводнике, импортировать новые ассеты (изображения, звуковые файлы и т.д.), импортировать/экспортировать пакеты (удобная возможность делиться готовыми файлами, в которых можно сохранить все взаимосвязи), поиск ссылок на ассет на сцене, выбор зависимых ассетов, обновление и реимпорт ассетов, открытие проекта в среде редактирования скриптов.

Hierarchy нужна для управления персонажами **GameObjects**. Все действующие персонажи игры отображаются в **Hierarchy**.

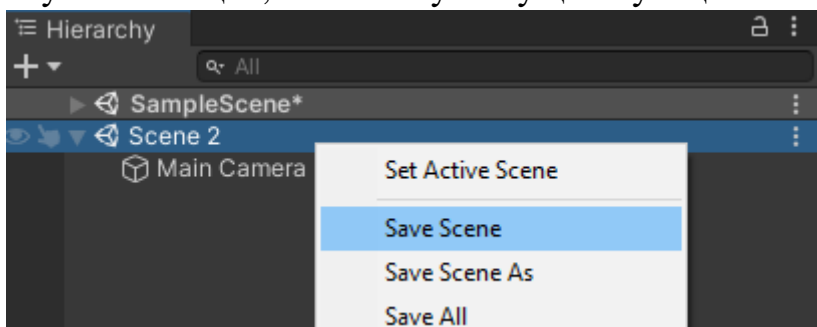
Hierarchy – окно иерархии отображает все объекты, которые находятся на сцене. В последних версиях Unity одновременно могут быть загружены несколько сцен для удобства копирования и перемещения объектов между ними. Для того, чтобы это реализовать, достаточно просто перетащить вторую сцену в окно иерархии или же с помощью нажатия ПКМ по нужной вам сцене.



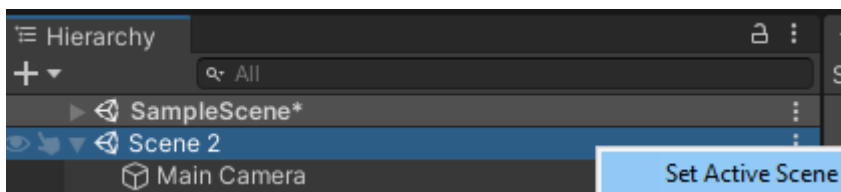
Сцены могут быть выгружены из памяти, но всё ещё находиться в окне иерархии, чтобы они не мешали работать, но в то же время были под рукой (нужно иметь минимум 2 сцены).



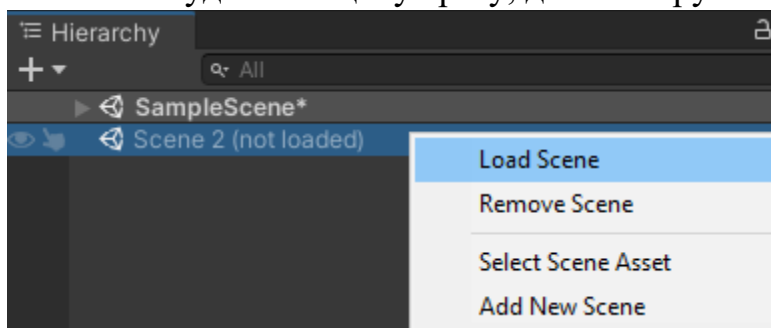
Сцены можно сохранять по-отдельности или все вместе сразу, также можно создавать новые файлы из полученных сцен, не ломая уже существующие.



Кнопка Set Active Scene позволяет вам переключаться между активными сценами, этот параметр указывает, в какой сцене у вас создаются объекты. Может быть только активная сцена.

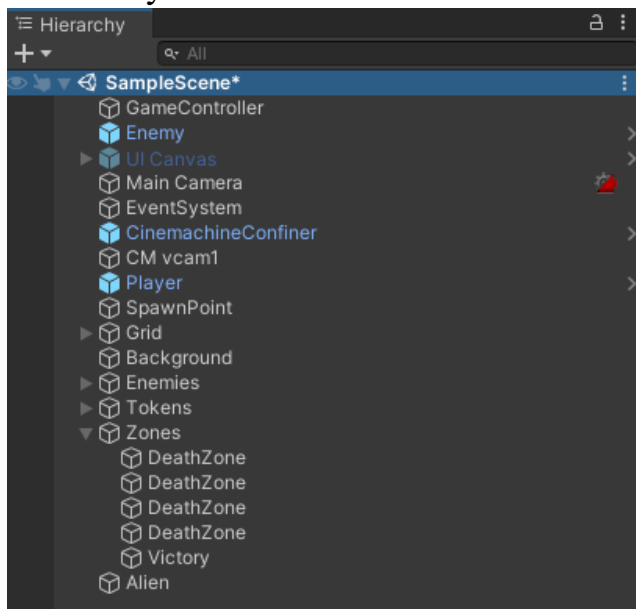


Когда вы выгрузили из памяти свою сцену, она может быть снова загружена или удалена, ничто не мешает вам удалить сцену сразу, до её выгрузки.



Проект, созданный в Unity может иметь достаточно много сцен, которые хранятся в соответствующем каталоге **Scenes** в окне **Project window**.

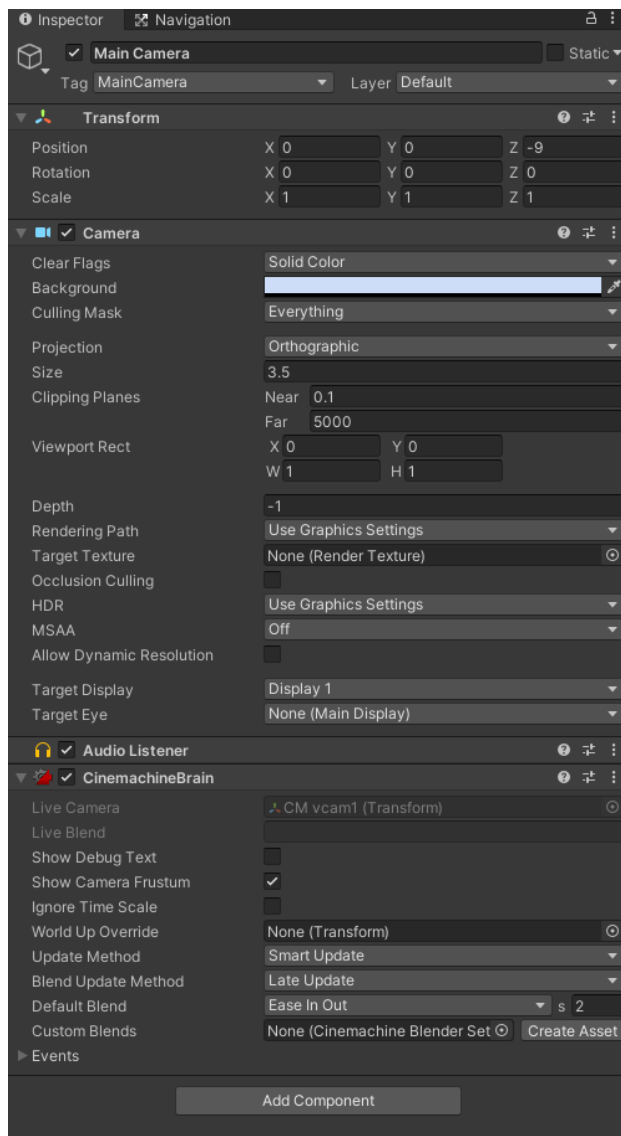
Игровые объекты в окне располагаются в определённой иерархии, более верхний объект является родительским. Значения позиции, поворота, масштаба зависят от родительских объектов. В окне иерархии находится кнопка создания объектов, поле поиска объектов по имени и типу.



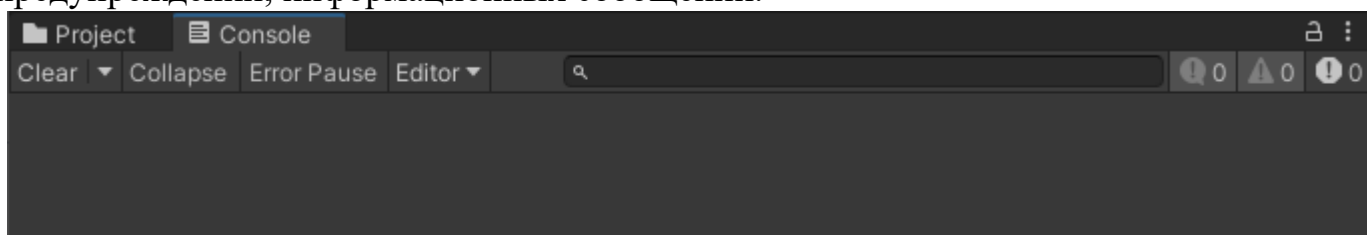
Отличие между **Project window** и **Hierarchy**: **Hierarchy** содержит все реально используемые на сцене игровые объекты, а **Project window** содержит все файлы ассетов, доступные для всего вашего проекта.

Inspector – окно инспектора отображает свойства, помогает настраивать параметры и поведение выбранного объекта или ассета.

- В верхней части инспектора отображается имя объекта
- Слева от имени переключатель, демонстрирующий активен ли объект
- Слева от переключателя для объекта может быть выбрана специальная иконка для отображения его на сцене (аналог Gizmo)
- Tag – для объекта может быть выбран особый тэг для удобства поиска и работы с объектом
- Layer – слой, к которому относится объект
- Справа от имени переключатель Static отвечает за включение всех статических настроек одновременно
- Далее выпадающий список, в котором можно произвести более точную настройку (настройки Static необходимы, например, для оптимизации).



Console – консоль, использующаяся для вывода сообщений об ошибках, предупреждений, информационных сообщений.



В верхней части окна находятся следующие кнопки:

- Clear (очистить консоль)
- Collapse (повторяющиеся сообщения отображать в виде одного)
- Clear on Play (очистить консоль при запуске, находится внутри Clear)
- Error Pause (ставить игру на паузу в редакторе при возникновении ошибки)

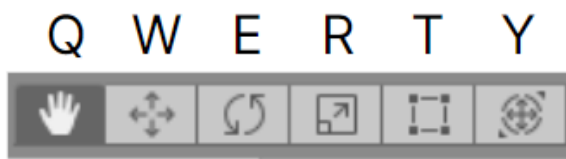
Последние три иконки обозначают количество информационных сообщений, предупреждений, ошибок (если они включены, то соответствующая информация отображается в консоли).

Toolbar (инструменты на верхней панели редактора) помогает передвигать персонажей (в т.ч. менять их размер и поворачивать), менять точку обзора, запускать и останавливать игру.

Инструменты на верхней панели редактора. К ним относятся несколько кнопок, переключателей, выпадающих списков с различными функциями.

Инструменты в левой верхней части:

- Hand tool (перемещение камеры сцены, горячая клавиша Q)
- Move tool (перемещение объектов, W)
- Rotate tool (поворот, E)
- Scale tool (масштабирование объекта, R)
- Rect Transform tool (позиционирование элементов интерфейса, может быть использован и для обычных объектов, T)
- Transform tool (смещение, масштабирование и вращение в одном, Y)
- Переключатель между центром и pivot-точкой (относительного чего выполняется поворот объекта, Z)
- Переключатель между глобальной и локальной системами координат (X).



Создавать **дубликаты объектов** можно с помощью комбинации клавиш Ctrl+D.

Инструменты в правой верхней части:

- Collab (Загрузить или отправить на сервер актуальную версию)
- Unity-сервисы
- Аккаунт пользователя
- Слои объектов
- Расположение окон.



НАВИГАЦИЯ ПО СЦЕНЕ

Для удобной навигации в окне сцены необходимо использовать мышь, клавиатуру, а также их сочетание:

ЛКМ – выделение объектов на сцене;

ПКМ – вращение камеры сцены;

Нажатие колесика мыши – перемещение камеры сцены (инструмент Pan);

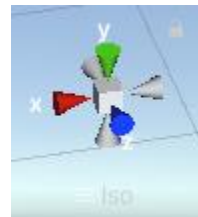
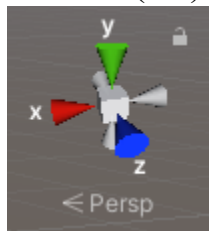
Колесико мыши – масштабирование камеры сцены;

Alt+ЛКМ – вращение камеры сцены вокруг активного объекта;

Alt+колесико мыши – масштабирование камеры сцены в определенную точку, соответствующую текущей позиции курсора;

ПКМ+клавиши WASD – движение камеры сцены по сцене (как в игре от первого лица).

В правой верхней части окна сцены располагается Scene Gizmo – инструмент, с помощью которого можно определить текущее положение камеры сцены относительно осей XYZ, а также переключаться между различными видами (сверху, снизу, справа, слева), в том числе изометрическим (Iso) и перспективным (Persp).



[В ОГЛАВЛЕНИЕ](#)

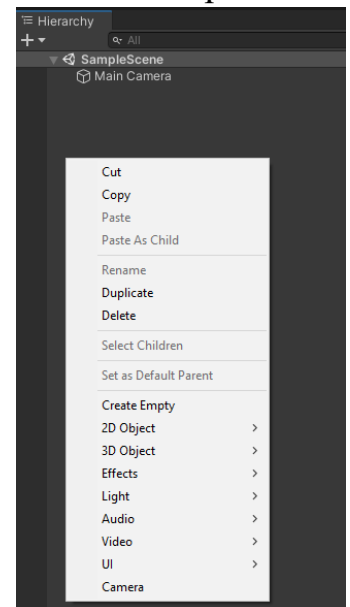
ИГРОВЫЕ ОБЪЕКТЫ

Игровой объект (**GameObject**) – это самый важный объект при разработке игр в Unity. Любой объект в Unity является игровым объектом. По сути, **GameObject** – это некий контейнер, который может содержать в себе различные компоненты, и в зависимости от них является или персонажем, или землёй, домом, автомобилем, роботом, каким-либо предметом инвентаря, элементом интерфейса, или вообще объектом, который не имеет визуального отображения.

Пустой игровой объект – также является важным понятием, так как этот **GameObject** не имеет внутри себя компонентов, кроме компоненты **Transform**. Пустые игровые объекты используются для формирования иерархии (структуры) объектов и совершения действий по отношению сразу к группе объектов.

Основные встроенные игровые объекты подразделяются на следующие категории:

- **Create Empty** – пустой игровой объект;
- **3D Object** – категория объектов, к которой относятся различные 3D-примитивы (куб, капсула, сфера и другие);
- **2D Object** – в данной категории доступны объекты **Sprite**, **Physics**, **Tilemap**, **Sprite Shape**, **Pixel Perfect Camera** и **Sprite Mask**;
- **Light** – источники света;
- **Audio** – объекты, связанные со звуком;
- **Video** – объекты, связанные с видео;
- **UI** – категория объектов для создания пользовательского интерфейса (UI);
- **Particle System** – система частиц (находится в **Effects**);
- **Camera** – камера.



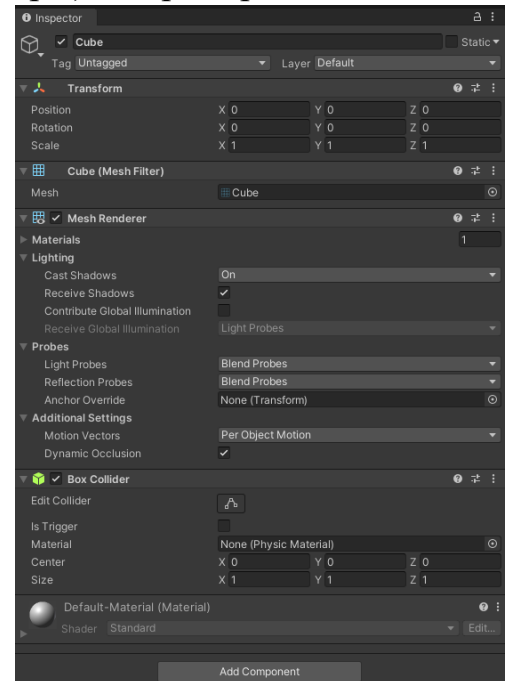
Доступ к ним осуществляется через меню **GameObject** на верхней панели редактора, через меню **Create** и контекстное меню в окне иерархии. Также игровые объекты можно создать через скрипт. Создаваемые игровые объекты содержат набор компонентов.

КОМПОНЕНТЫ

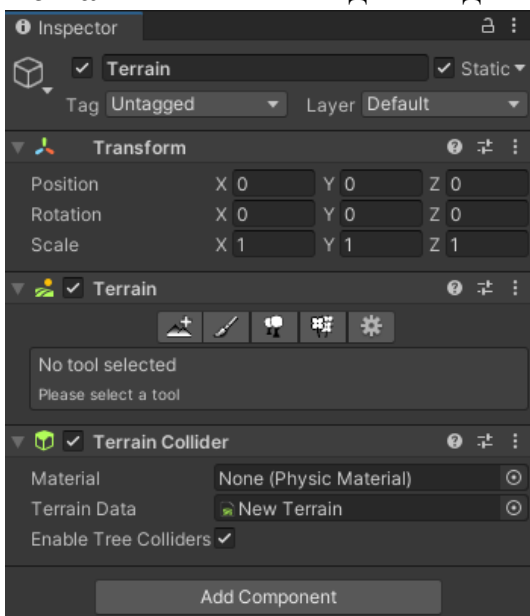
Компоненты являются составляющими частями любого игрового объекта, они придают ему определённый набор свойств и поведение. Обязательным компонентом, который включается по умолчанию для любого объекта, является компонент Transform (для элементов интерфейса Rect Transform).

Некоторые основные компоненты и их свойства (параметры) на примере **Cube**:

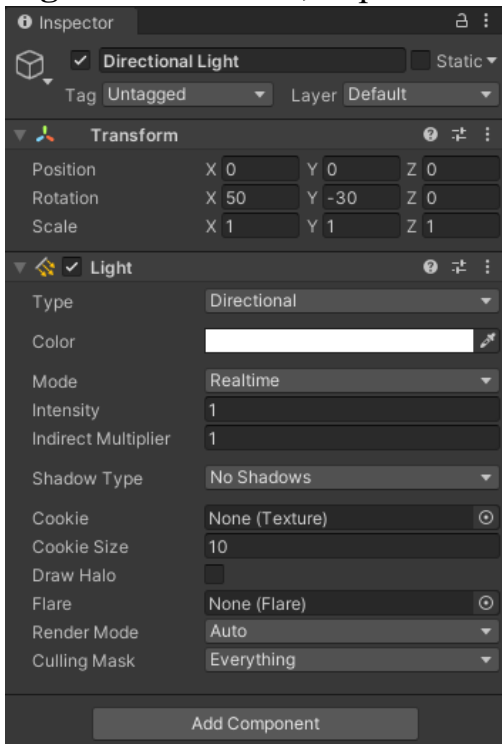
- Transform – отвечает за положение объекта на сцене (Position), поворот (Rotation), масштаб (Scale);
- Mesh Filter – передает mesh (набор вершин и многоугольников, определяющих форму объекта) в Mesh Renderer для отрисовки;
- Box Collider – позволяет объекту взаимодействовать с другими объектами;
- Mesh Renderer – компонент отрисовки mesh-а;



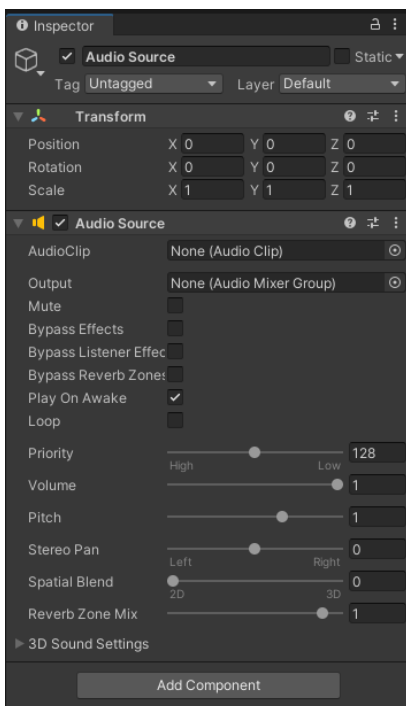
Terrain – компонент для создания поверхности, рельефа;



Light – компонент, определяющий тип источника света, цвет, интенсивность и т.д.;



Audio Source – компонент для настройки и проигрывания 2D и 3D-звуков. Данный компонент является «источником» звука, на сцене также необходим «слушатель», объект с компонентом **Audio Listener**, обычно он по умолчанию находится на камере.



Компонент может быть добавлен к объекту:

- Кнопка Add Component в нижней части окна инспектора;
- Через меню Component в верхней части редактора;
- Перетащить ЛКМ из окна проекта на объект в окне иерархии или в окно инспектора (определенные типы ассетов не могут быть добавлены в качестве компонентов таким образом);
- Через скрипт.

[В ОГЛАВЛЕНИЕ](#)

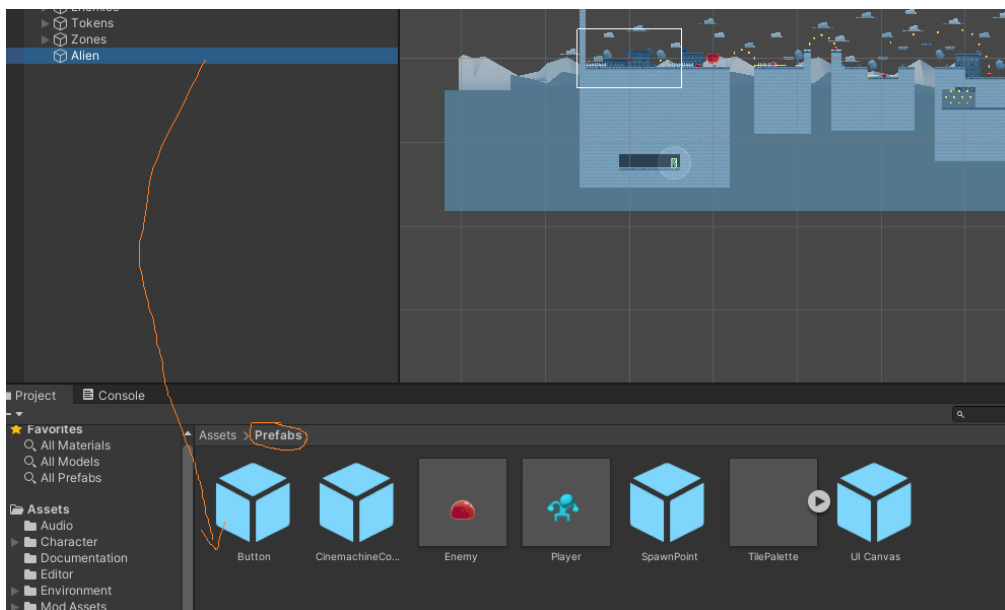
ПРЕФАБЫ

Prefab – особый тип ассетов, позволяющий хранить весь игровой объект со всеми компонентами и значениями их свойств. Это некий шаблон для создания экземпляра хранимого объекта на сцене. Любые изменения в префабе отражаются на его экземплярах, но при этом для каждого экземпляра в отдельности можно переопределить компоненты, не применяя изменения ко всем другим экземплярам.

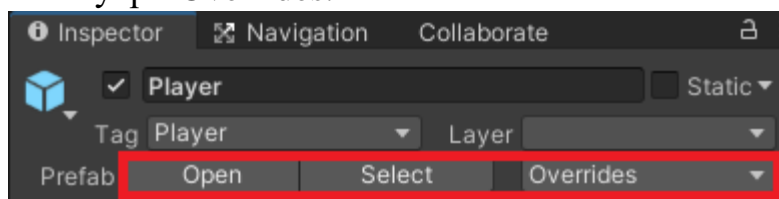
У нас есть некий объект на сцене, который повторяется множество раз, для этой цели удобно использовать префабы и при необходимости применять изменения ко всем экземплярам префаба одновременно. Например, в игре Марио: монетки, трубы, кирпичи – всё это могло бы быть префабами.

Создать префаб можно следующим образом:

- Assets – Create – Prefab (верхнее меню редактора);
- Create – Prefab (контекстное меню в окне проекта);
- Перетащить текущий GameObject из окна иерархии в необходимую папку окна проекта.



В окне иерархии префаб выделен синим цветом, а в окне инспектора у него появляются дополнительные кнопки: Select (для выбора этого префаба в окне проекта), Revert (отмена изменений у текущего экземпляра префаба до состояния исходного префаба, но это не касается позиции и поворота объекта), Apply (применение изменений текущего экземпляра префаба ко всем экземплярам, в том числе исходному префабу). Последние две кнопки находятся внутри Overrides.

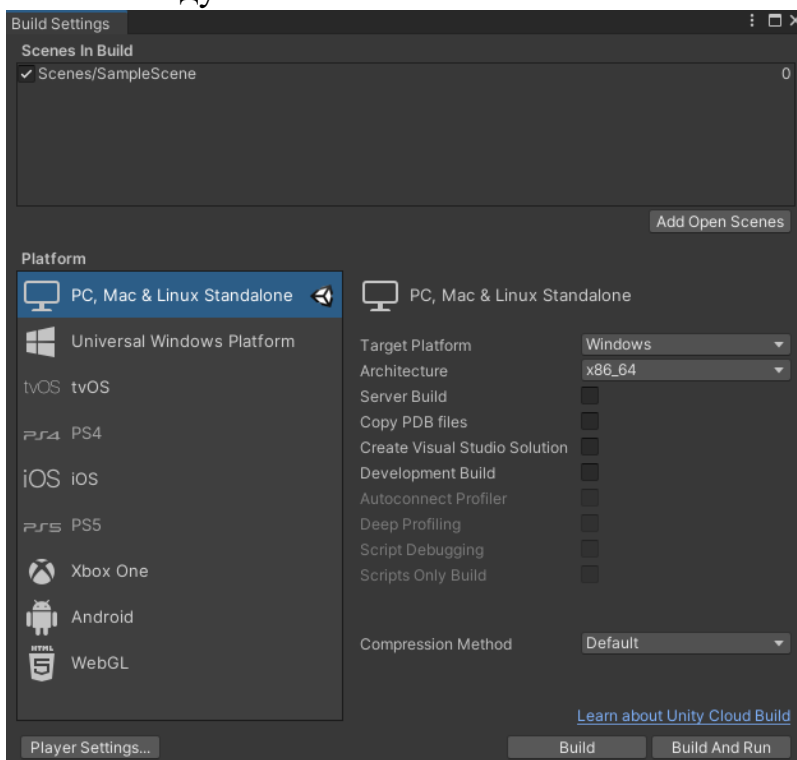


Для того, чтобы объект на сцене перестал являться префабом и, соответственно, потерял все зависимости, необходимо выделить его в окне иерархии и нажать GameObject – Break Prefab Instance.

[В ОГЛАВЛЕНИЕ](#)

СБОРКА ПРОЕКТА

Для того чтобы собрать проект под определённую платформу, необходимо на верхней панели редактора выбрать File – Build Settings. Текущая выбранная платформа отмечена значком Unity. Для переключения необходимо выбрать другую платформу ЛКМ и нажать Switch Platform. Некоторые платформы требуют загрузки и установки дополнительных SDK или модулей.



По умолчанию проект настроен для сборки проекта под «PC, Mac & Linux Standalone». Убедитесь, что необходимые сцены добавлены в верхней части окна Scenes In Build. Для того, чтобы добавить сцены, которые войдут в сборку, надо их просто перетащить в окно Scenes in Build или нажать кнопку Add Open Scenes, чтобы добавить открытые сцены. Далее необходимо нажать Build, выбрать расположение и название исполняемого файла.

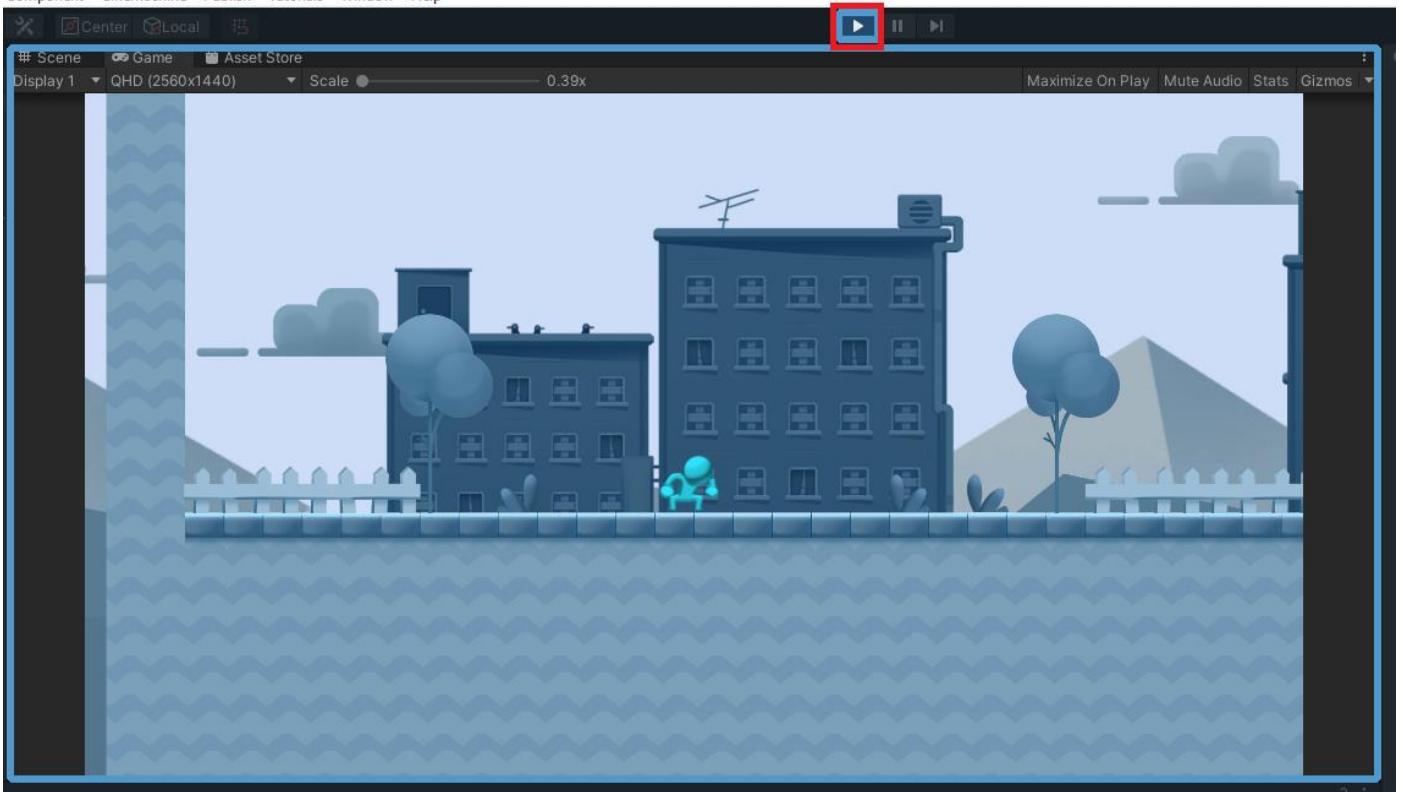
В дальнейшем, когда все настройки заданы, можно сразу нажимать Build And Run в меню File – Build And Run. В таком случае, после того как проект соберётся, он автоматически запустится.

[В ОГЛАВЛЕНИЕ](#)

ПРИЛОЖЕНИЕ 1. ТУТОРИАЛЫ К 2D PLATFORMER

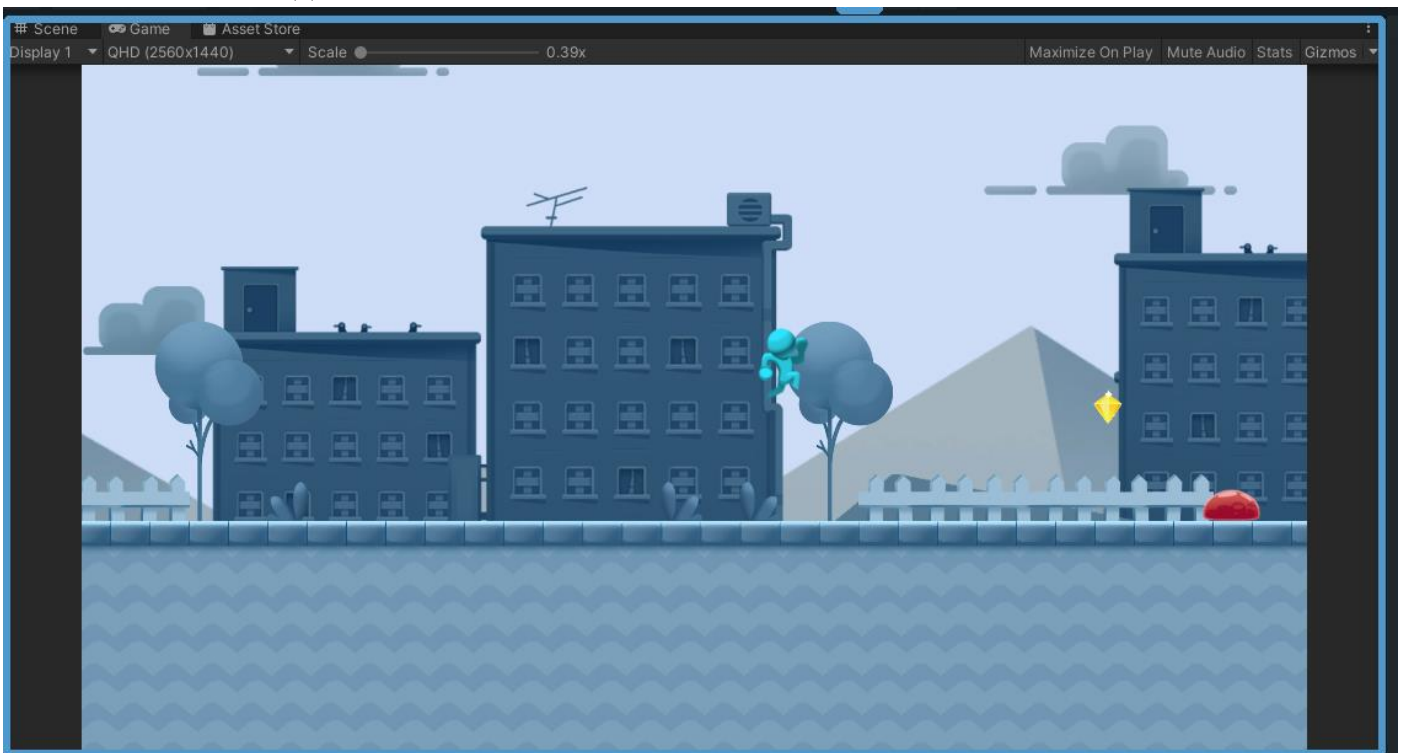
TUTORIAL 1: GET STARTED

Пробуем игровой режим нажатием по белому треугольнику (Play mode):

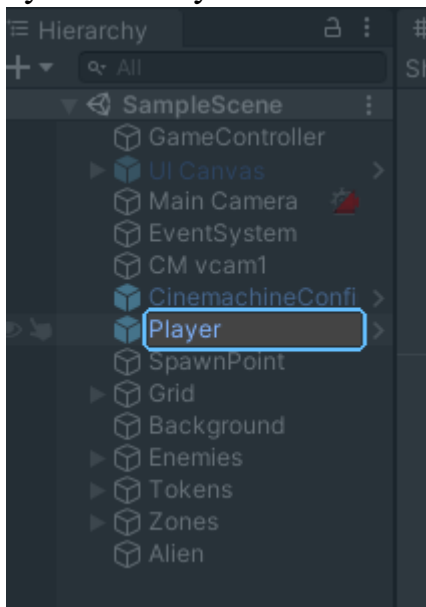


Управляем персонажем с помощью WASD (именно удержанием кнопок), прыгаем с помощью клавиши Пробел. По окончании тестов нажимаем вновь на белый треугольник.

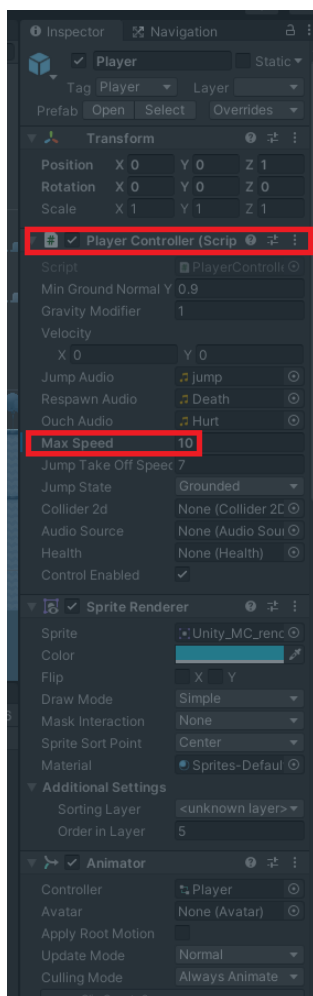
Внимание! Лучше приглушите колонки или наушники, от используемой в игре музыки может немного подташнивать.



Все объекты игры называются в Unity **GameObject**. Их общий список (**Hierarchy**, иерархия) представлен в левой части экрана редактора. Чтобы изменить параметры персонажа Player можно щелкнуть по нему в левой части.



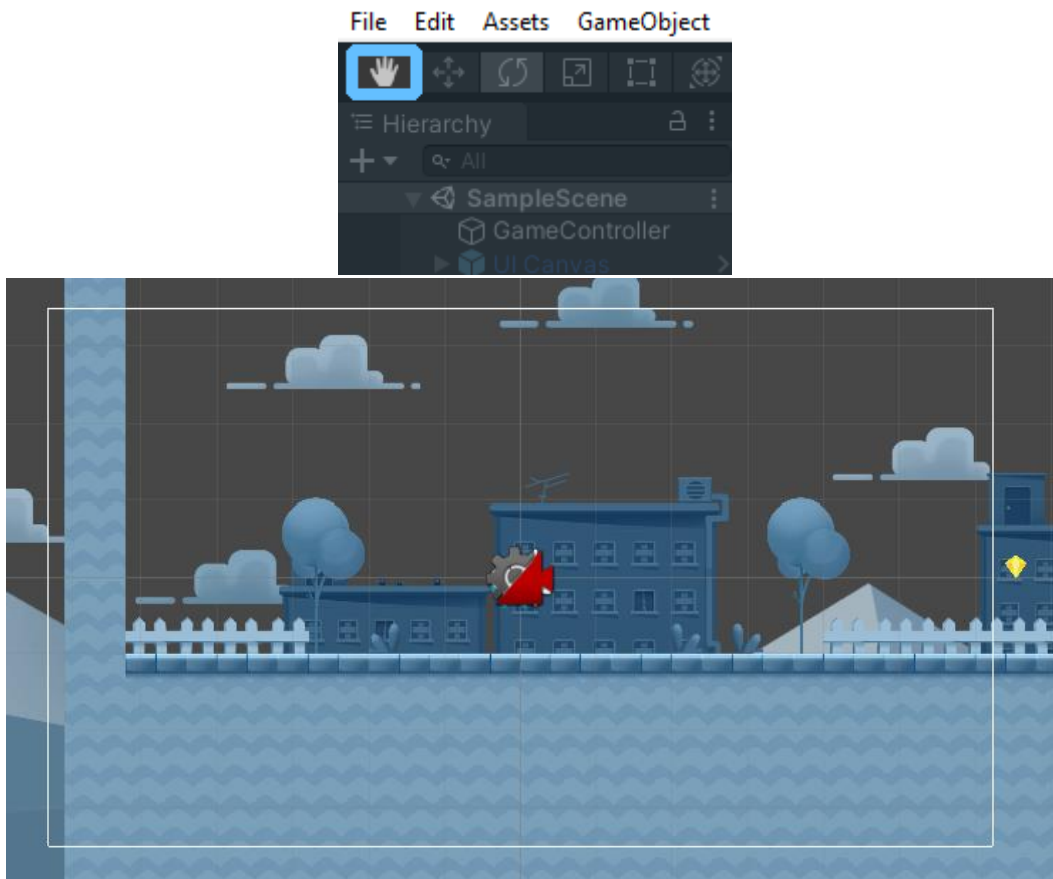
В правой части экрана появится куча настроек персонажа Player (окно **Inspector**). В частности, вкладка **Player Controller** отвечает за движение персонажа в заданное игроком соответствующее направление. В данном туториале от нас требуется изменить параметр максимальная скорость (**max speed**) на 10. Сохранение проекта осуществляется комбинацией **Ctrl + S** или на панели **File → Save**. Завершили **TUTORIAL 1**.



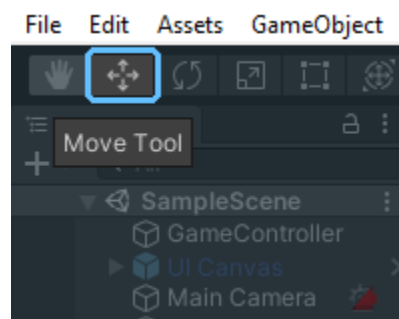
TUTORIAL 2: EDITOR BASICS

Обучимся использованию инструментов **Hand Tool** (изменение обзора) и **Move Tool** (перемещение **GameObjects**).

Используем **Hand Tool**: ЛКМ для перетягивания сцены и ее содержимого, колесико для масштабирования.



Используем **Move Tool**. ЛКМ для перемещения персонажа (**GameObject**). При большом количестве персонажей удобнее их перемещать с помощью этого инструмента, нежели с использованием **Hierarchy**.

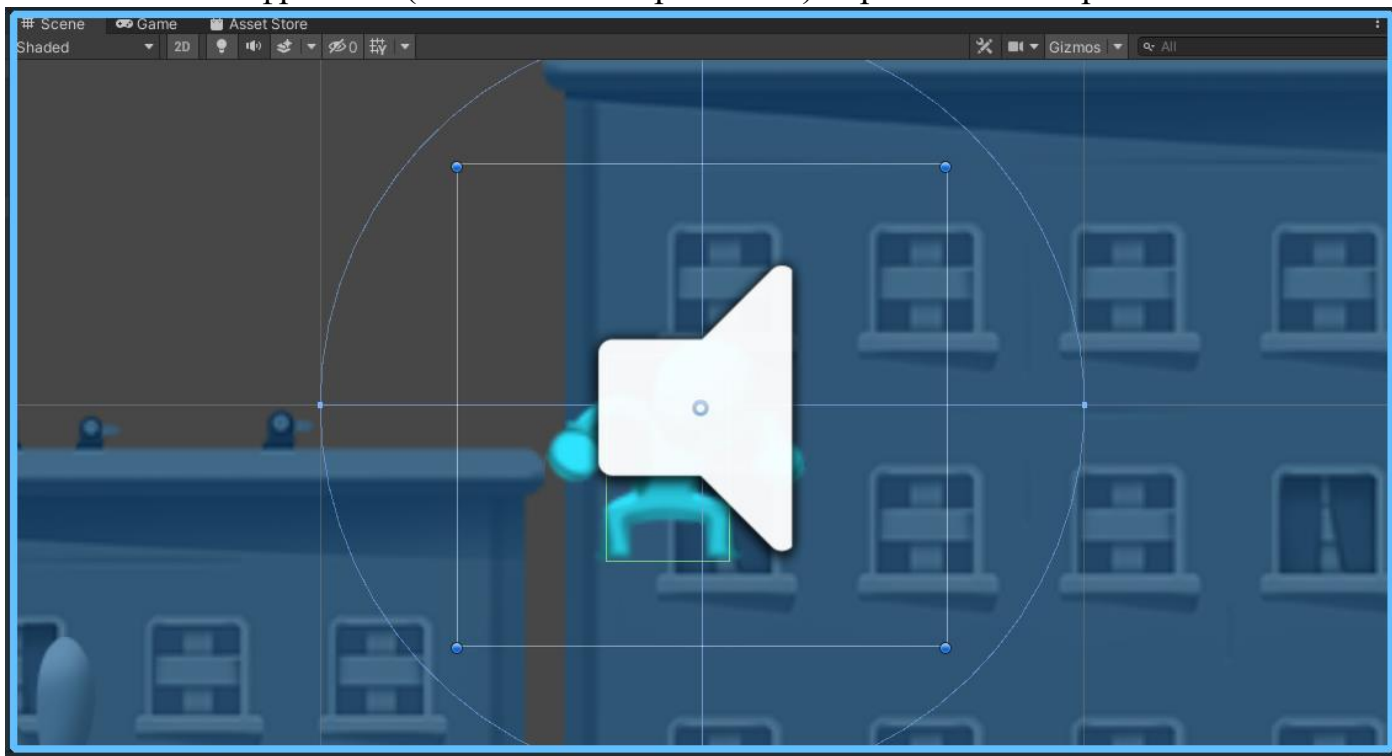


Не забываем сохраниться (Ctrl + S). Завершили **TUTORIAL 2**.

TUTORIAL 3: CHANGE COLORS

Для изменения цвета персонажа на сцене необходимо сначала выбрать его ЛКМ в **Hierarchy** слева.

Поскольку карта сцены может быть очень большой или персонаж может быть сам по себе очень маленьким, воспользуйтесь клавишей **F** на клавиатуре. Эта кнопка максимально зафреймит (максимально приблизит) персонажа на экране.



Цвет персонажа меняется справа в открывшейся вкладке **Inspector**, в блоке **Sprite Renderer**, пункт **Color**. Изменяем цвет, включаем игру, сохраняем ее через **Ctrl + S**.

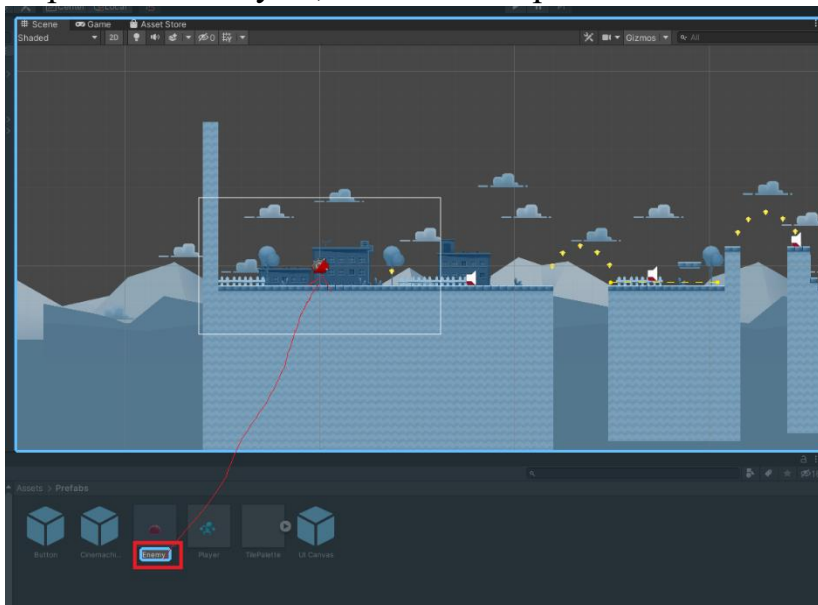


Завершили **TUTORIAL 3**.

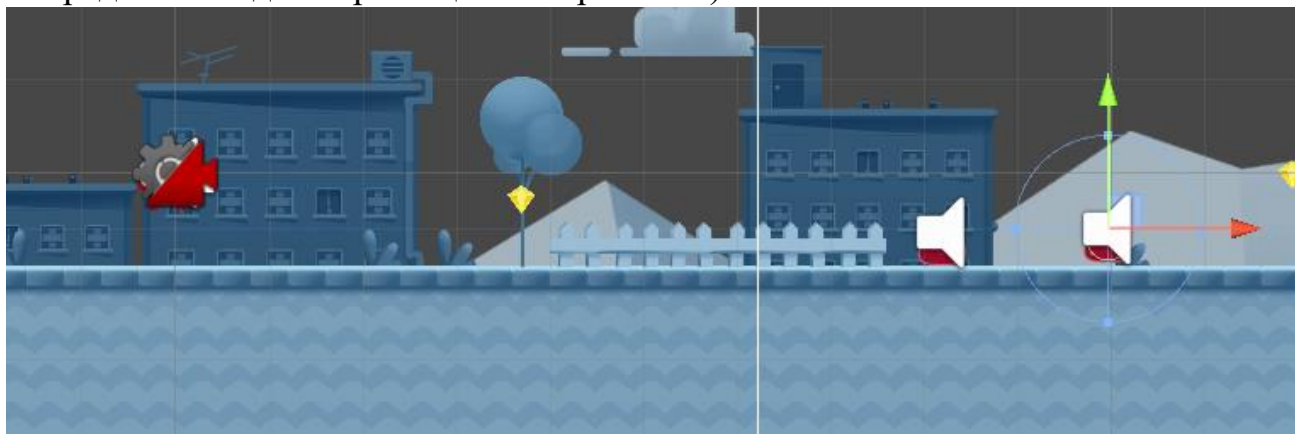
TUTORIAL 4: ADD AN ENEMY

Учимся добавлять **Prefab** врага в игру.

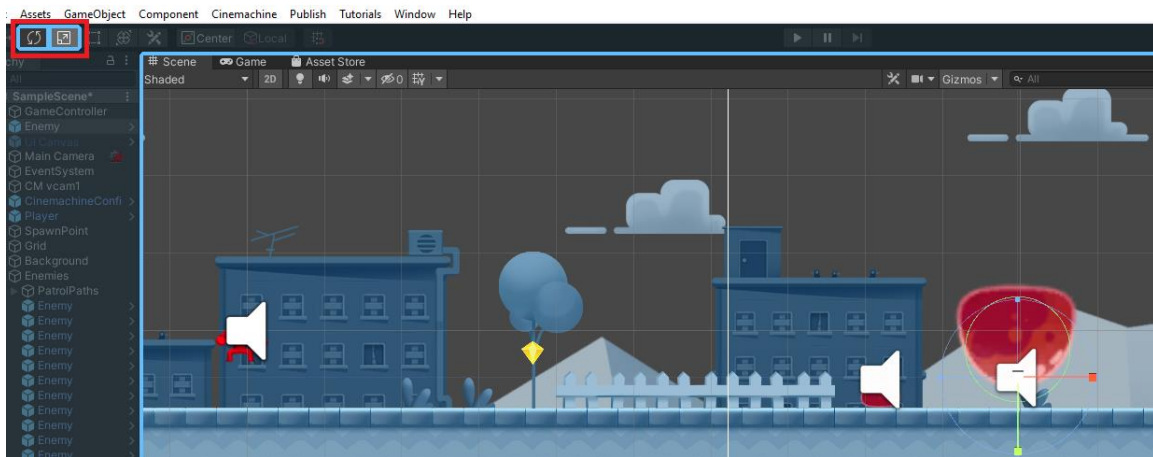
Prefab (Префаб) – это **asset** (пакет), который действует как шаблон **GameObject**. Из префаба можно создать несколько копий, называемых экземплярами (**instances**). Изменение префаба приводит к изменению всех его экземпляров. Например, чтобы превратить много деревьев в кактусы, вы можете просто изменить один префаб дерева!



Включите **MoveTool** (клавиша **W**) и поставьте врага в удобное для вас место. Для удобства (в плане обзора) можно пользоваться стрелками клавиатуры или ЛКМ (непосредственно для перемещения персонажа).



Персонажа можно вращать (**Rotate Tool**, клавиша **E**) или изменять его размер (**Scale Tool**, клавиша **R**).

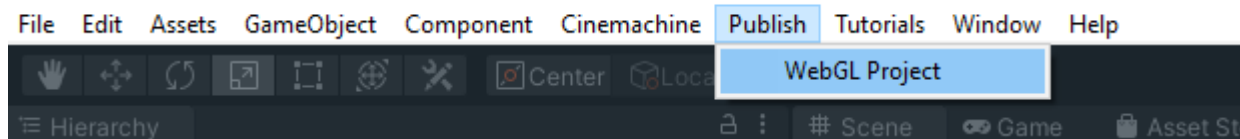


Завершили **TUTORIAL 4.**

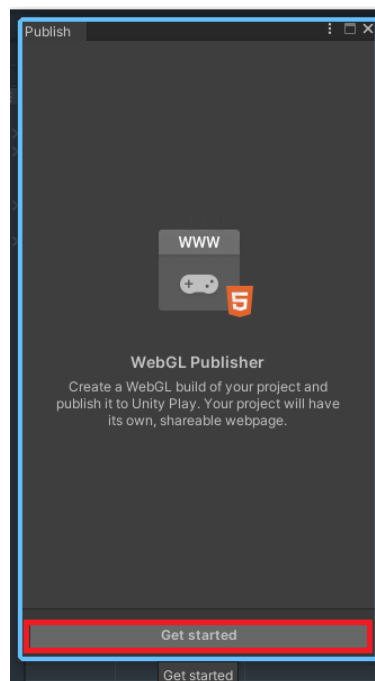
TUTORIAL 5: BUILD AND PUBLISH (ознакомиться!)

Экспортируем проект микроигры как WebGL на платформе Unity Play.

2D Platformer FIIT - SampleScene - PC, Mac & Linux Standalone - Unity 2020.3.16f1 Personal <DX11>

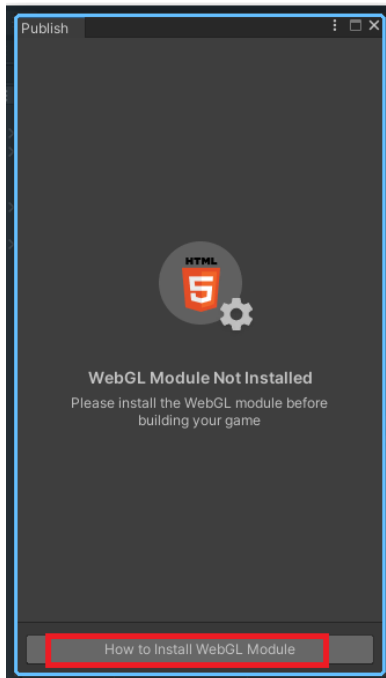


Нажимаем «Get started».



Для дальнейшей работы потребуется [установка WebGL модуля](#).

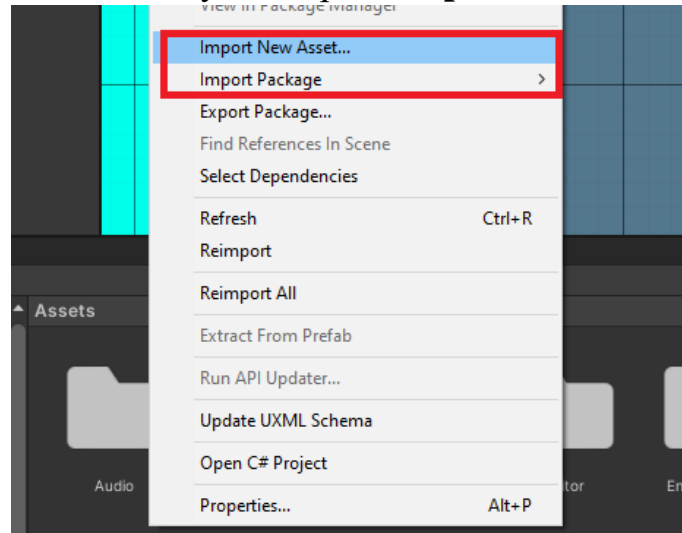
Перед публикацией игры потребуется создать исполняемый файл для запуска игры с компьютера. Этот шаг может занять 30 минут.



P.S. Возможно здесь сыграет свою роль тот факт, что мы в свое время не установили крайнюю версию **2021.3.30f1**, и поэтому пакет WebGL доустановить не получится 😊
Тutorials по этому проекту кончились!
Если вы захотите еще полазить в нем, рекомендуем перейти [на эту страницу](#).

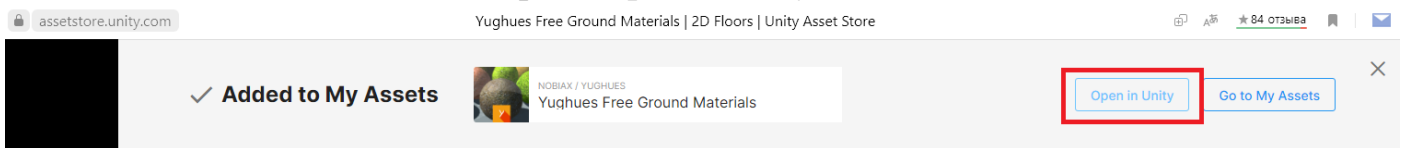
ПРИЛОЖЕНИЕ 2: ДОБАВЛЕНИЕ ASSETS И PACKAGES

Первый способ добавления ассета или пакета удобен в случае, если у вас на компьютере уже есть необходимые для загрузки в проект файлы. В этом случае необходимо лишь щелкнуть ПКМ в окне Assets внизу и выбрать **Import New Asset** или **Import Package**.

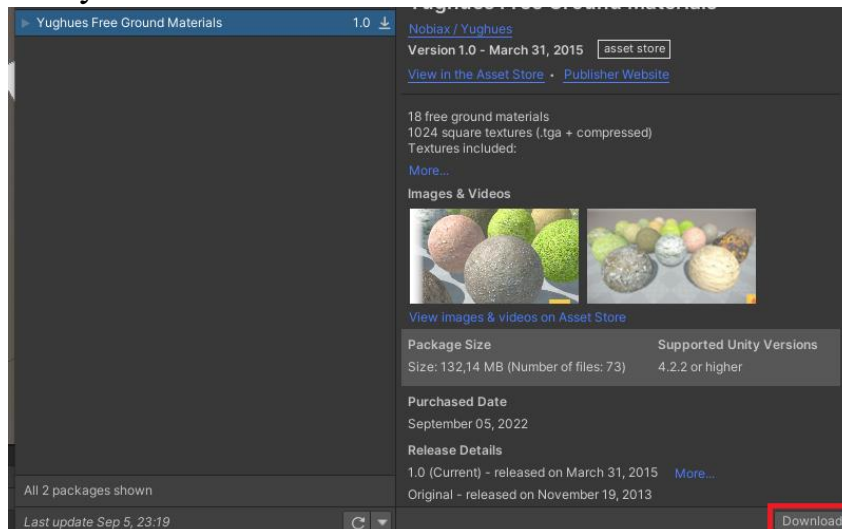


Чтобы открыть Asset Store раньше можно было воспользоваться вкладкой Window → Asset Store прямо из-под Unity Editor. Теперь, если вы хотите добавить **asset** из [Unity Asset Store](#), необходимо открыть сайт и залогиниться на нем с помощью Unity ID. Рассмотрим добавление asset`а материала с названием Yughues Free Ground Materials.

1. В окно поиска вверху вбейте materials.
2. Справа в разделе Pricing выберите Free Assets.
3. Выберите Yughues Free Ground Materials
4. Нажмите Add to my Assets.
5. В появившемся окне выберите Open in Unity.

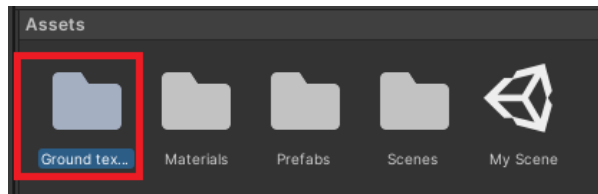


6. Нажмите кнопку Download.



7. Нажмите кнопку Import.

8. В папке Assets появится новая папка.



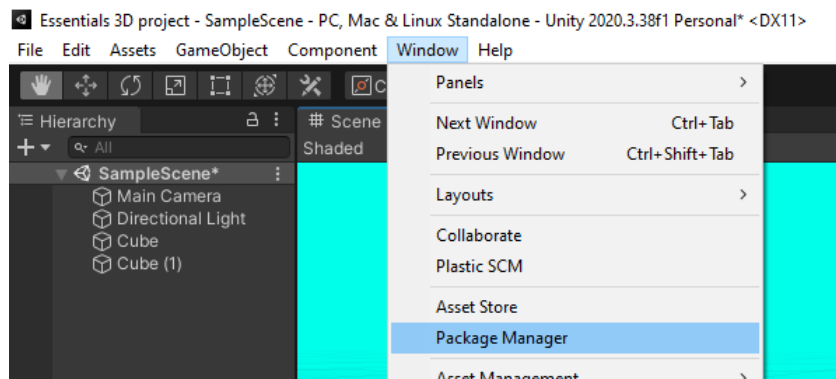
Тutorial по добавлению пакетов есть [здесь](#).

Для добавления пакетов необходимо познакомиться с **Package Manager**. Более подробно об этом можно [узнать из мануала](#).

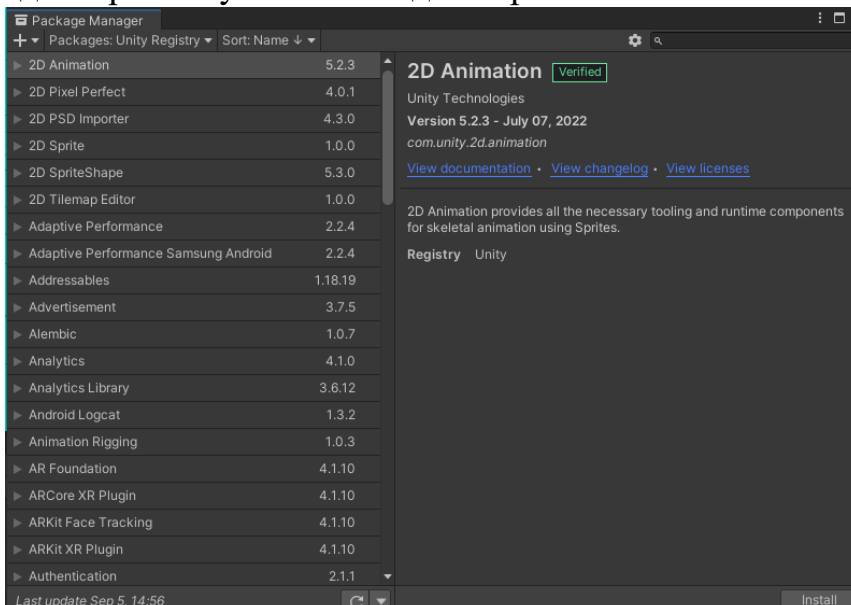
Пакеты – это небольшие наборы фич или ассетов Unity, которые помогают делать разные вещи в ваших проектах. К ним относятся:

- Основные функции Unity, которые по умолчанию устанавливаются вместе с Unity Editor.
- Дополнительные пакеты, которые вы можете установить в проект, когда они вам понадобятся.

Для установки пакетов рекомендуется использовать **Package Manager: Window** → Package Manager.

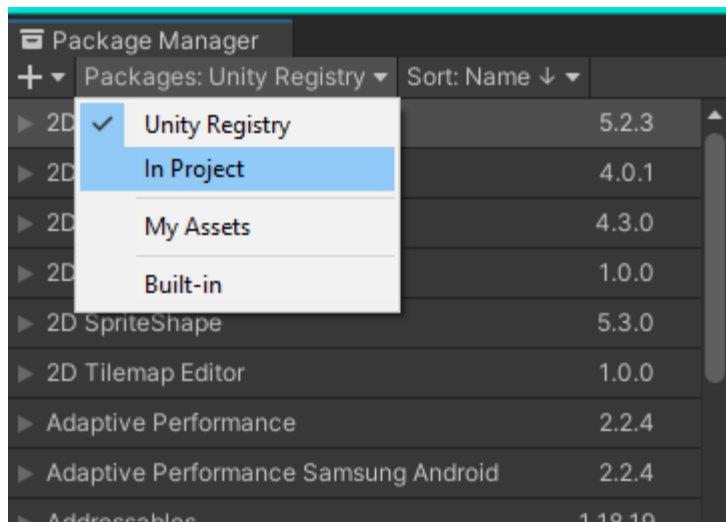


По умолчанию здесь будут представлены все доступные пакеты, которые можно использовать в редакторе Unity. Пакетов здесь приличное количество.

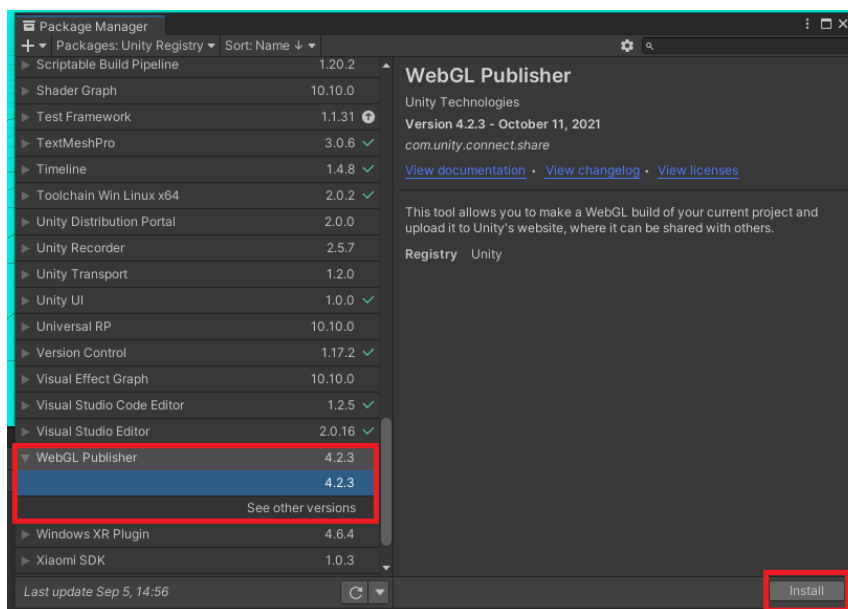


Правило хорошего тона предполагает постепенное добавление пакетов и ассетов в ходе работы над проектом вместо добавления «всего и сразу», что многократно увеличивает размер итоговой игры. Информация о каждом пакете может быть найдена [тут](#).

Чтобы в указанном выше окне отобразить пакеты, используемые в текущем проекте, выберите In project.

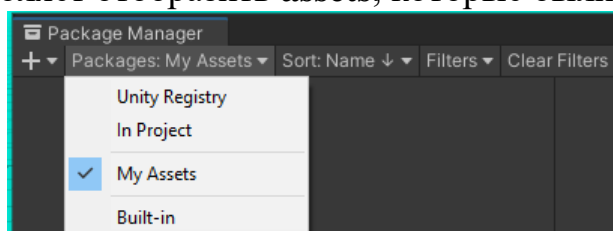


Например, для «расшаривания» игры необходимо подключить Web-GL пакет (необязательно).



После установки он появится на вкладке In project.

Вкладка My Assets позволяет отобразить **assets**, которые были импортированы извне.



ДОМАШНЕЕ ЗАДАНИЕ

1. Ознакомиться с информацией из данного файла.
2. Установить Unity Hub и Unity Editor на свой ноутбук / домашний ПК.
3. Запустить обучающий проект 2D Platformer через Unity Hub (см. подсказку выше).
4. Пройти по туториалам из [Приложения 1](#).
5. В проекте 2D Platformer найти объекты, которые отвечают за игровую логику и управление игроком, сделать следующие изменения:
 - a. Максимальная скорость игрока 6;
 - b. Найти скрипт, отвечающий за столкновение игрока и врага. Модернизировать игру таким образом, чтобы игрок не умирал при столкновении с врагом (уменьшением урона от врага/увеличением HP игрока/изменением логики в коде).
 - c. Отрегулировать амплитуду прыжка игрока (чтобы мог прыгать выше даже коротким нажатием клавиши пробел).
 - d. Измените спаун игрока после смерти на 10;
 - e. Замените звуковые эффекты на свое усмотрение
 - f. Попробуйте сделать так, чтобы спрайт Token крутился в обратную сторону.

ИНТЕРЕСНЫЕ ПРОЕКТЫ

- 2D
 - [Рогалик от Unity \(2D\)](#)
 - [Изучение Unity через RPG \(2D\)](#)
 - [Машинка \(2D, Android\)](#)
 - [Платформенный теннис а.к.а. аэрохоккей \(2D\)](#)
 - [2D Platformer](#)
 - [Space Shooter \(2D\)](#)
 - [Еще один Space Shooter \(2D\)](#)
 - [2D игра на Unity. Подробное руководство.](#)
- 3D
 - [Про 3D игры от Unity](#)
 - [Обход препятствий \(3D\)](#)
 - [Multiplayer FPS \(3D\)](#)
 - [Tower Defense \(3D\)](#)
 - [RPG in Unity \(3D\)](#)

[В ОГЛАВЛЕНИЕ](#)

СПИСОК ЛИТЕРАТУРЫ

Интернет-источники

1. Официальный [сайт Unity](#).
2. Документация по Unity 2020.3 ([на русском](#) (2020.2) и [английском](#)).
3. [Unity Learn](#): платформа с обучающими туториалами и различными уроками.
4. [Unity Asset Store](#).
5. [Brackeys](#).
6. [Skillbox](#). Как создать игру на Unity.
7. [НОУ Интуит](#). Разработка игр на Unity.
8. [Введение в Unity 2D](#).
9. [A curated list of awesome Unity Games \(Open Source\), code examples and resources](#).
10. [120+ книг по Разработке Игр. Unity и геймдев в общем](#).

Книги

1. Бонд Д. Г. Unity и C#. Геймдев от идеи до реализации. 2-е изд. – СПб.: Питер, 2019. – 928 с.: ил. – (Серия «Для профессионалов»).
2. Мэннинг Д., Батфилд-Эддисон П. Unity для разработчика. Мобильные мультиплатформенные игры. – СПб.: Питер, 2018. – 304 с.: ил. – (Серия «Бестселлеры O'Reilly»).
3. Торн А. Искусство создания сценариев в Unity / пер. с англ. Р. Н. Рагимова. – М.: ДМК Пресс, 2016. – 360 с.: ил.
4. Хокинг Д. Unity в действии. Мультиплатформенная разработка на C#. 2-е межд. изд. – СПб.: Питер, 2019. – 352 с.: ил. – (Серия «Для профессионалов»).
5. Quick J. M. Learn to Code with Games. – CRC Press, 2015.
6. Quick J. M. Learn to Implement Games with Code. – AK Peters/CRC Press, 2016.
7. Godbold A. Mastering UI Development with Unity: An in-depth guide to developing engaging user interfaces with Unity 5, Unity 2017, and Unity 2018. – Packt Publishing Ltd, 2018.
8. Godbold A., Jackson S. Mastering Unity 2D Game Development. – Packt Publishing Ltd, 2016.

УЧЕБНАЯ КАРТА ДИСЦИПЛИНЫ

Разработка игр

Задание	Баллы
Знакомство с редактором Unity и реализация игры Roguelike	10
Разработка базового варианта игры Arcanoid	15
Разработка расширенного варианта игры Arcanoid (выполнение дополнительных заданий)	15
Разработка базового варианта тренажера English Words с реализацией этого варианта для мобильной платформы или разработка 3D-игры	30
Итого	70

Дополнительные наборы бонусных заданий из электронного задачника Programming Taskbook

Одно задание по теме «Паттерны ООП» из задачника Programming Taskbook for OOP – 10 баллов.

Набор из шести заданий по теме «**Linq to XML**» из задачника Programming Taskbook for LINQ (группа LinqXml); пять заданий — по 4 балла, шестое задание – 10 баллов.

Конкретные наборы бонусных заданий генерируются автоматически для каждого студента.

Последний срок сдачи бонусных заданий — 25 декабря 2023 г.

Общие требования

В течение семестра студент должен набрать не менее 38 основных (не бонусных) баллов!

Если суммарное количество баллов, набранных в течение семестра (с учетом бонусных), **больше 84**, то студент автоматически получает оценку «отлично» за экзамен, с выставлением полученного числа баллов. Если суммарное количество баллов, набранных в течение семестра (с учетом бонусных), больше или равно 60 (но меньше 85), то *по желанию студента* это количество баллов может считаться итоговым с выставлением соответствующей оценки за экзамен, без сдачи самого экзамена. В случае отсутствия такого желания, студент **сдает экзамен**, но бонусные баллы учитываются в *половинном* размере.

Если суммарное количество баллов, набранных в течение семестра (с учетом бонусных), **меньше 60** (и количество основных баллов больше 37), то студент **сдает экзамен** и при условии успешной сдачи экзамена (набора на экзамене не менее 22 баллов) получает оценку, определяемую суммой всех баллов, набранных в течение семестра (бонусные баллы учитываются в *половинном* размере), и баллов, набранных на экзамене.

Экзамен проводится в форме теста в системе Moodle с вопросами по основным возможностям языка C# (работа с коллекциями, строками, файлами; объектная модель C#, структуры, классы и интерфейсы, обобщенные классы, наследование, полиморфизм; делегаты и лямбда-выражения, последовательности). Все эти вопросы будут обсуждаться в заключительной части лекционного курса.

Баллы за семестр	Экзамен
Менее 38 основных (не бонусных) баллов	Не допущен к сдаче экзамена, необходим добор баллов.
38-59 баллов, в том числе не менее 38 основных баллов	Сдача экзамена; бонусные баллы, набранные в течение семестра, учитываются в <i>половинном</i> размере.
60-84 балла, в том числе не менее 38 основных баллов	Вариант 1. Оценка «удовлетворительно» или «хорошо» без сдачи экзамена (при согласии студента)
	Вариант 2. Сдача экзамена; бонусные баллы, набранные в течение семестра, учитываются в <i>половинном</i> размере.
85 баллов и более	Оценка «отлично» без сдачи экзамена

[В ОГЛАВЛЕНИЕ](#)